

A DYNAMIC SYSTEMS ENGINEERING METHODOLOGY RESEARCH STUDY

PHASE TWO:

**"EVALUATING METHODOLOGIES, TOOLS, AND TECHNIQUES FOR
APPLICABILITY TO NASA'S SYSTEMS PROJECTS"**

Submitted to:

**NATIONAL AERONAUTICS AND SPACE ADMINISTRATION
GODDARD SPACE FLIGHT CENTER
GREENBELT MARYLAND 20770**

Submitted by:

**SCHOOL OF ENGINEERING, HOWARD UNIVERSITY
WASHINGTON, D.C. 20059**

(NASA-CR-181319) A DYNAMIC SYSTEMS
ENGINEERING METHODOLOGY RESEARCH STUDY.
PHASE 2: EVALUATING METHODOLOGIES, TOOLS,
AND TECHNIQUES FOR APPLICABILITY TO NASA'S
SYSTEMS PROJECTS (Howard Univ.) 145 p

N90-14985

Unclass

63/81 0252530

A DYNAMIC SYSTEMS ENGINEERING METHODOLOGY RESEARCH STUDY

PHASE TWO:

**"EVALUATING METHODOLOGIES, TOOLS, AND TECHNIQUES FOR
APPLICABILITY TO NASA'S SYSTEMS PROJECTS"**

Submitted to:

**NATIONAL AERONAUTICS AND SPACE ADMINISTRATION
GODDARD SPACE FLIGHT CENTER
GREENBELT MARYLAND 20770**

Submitted by:

**SCHOOL OF ENGINEERING, HOWARD UNIVERSITY
WASHINGTON, D.C. 20059**

Prepared by:

**Arthur S. Paul, Ph.D
Tepper L. Gill, Ph.D
and
Arlene P. Maclin, Ph.D**

October 10, 1989

PREFACE

This Report presents the purpose, approach, and results of Phase Two of a research grant to Howard University (Grant Number NAG 5-995) from the National Aeronautics and Space Administration (NASA).

The purpose of the grant is to study systems engineering methodologies in light of changing environments and changing needs. The results of this investigation are to be used to identify and validate new methodologies with potential applications to NASA's systems life-cycle planning processes.

The study was designed to have two phases: Phase One, completed in November 1988 was a study of NASA's systems projects and the need for systems engineering methodologies; and Phase Two, this phase, involved evaluating methodologies, tools, and techniques with potential for application to NASA's systems projects, and making recommendations to NASA.

This study is sponsored and managed by the Networks Division (ND) of the Mission Operations and Data Systems Directorate (MO&DSD). The primary methodology being used by the Division is described in the MO&DSD Systems Management Policy. This methodology, which has been developed for Directorate-wide application, has been evaluated with six others from government and industry for applicability to the ND's projects.

ACKNOWLEDGMENTS

The authors would like to thank the staff of the Networks Division, particularly the Network Control Systems Branch, for its cooperation and assistance throughout the effort. Special thanks to Mr. Adolph Goodson, National Aeronautics and Space Administration's (NASA's) Technical Officer, for his resourcefulness and dedication to the project.

Thanks to Mr. Lee Saegesser, NASA's History Division and Ms. Monika Montgomery-Clingman of the Educational Publication Distribution Center for their assistance with information, including the identification of some not-so-obvious sources of data.

Thanks also to the many speakers from industry and academia who participated and contributed to a series of very lively and enlightening seminars at Howard University and at Goddard Space Flight Center. A list of presenters for this phase of the project is presented in Appendix A.

Very special thanks to Computer Sciences Corporation, Defense Systems Management College, Jet Propulsion Laboratory, HUGHES, Integrated Computer Systems, and the Center for Interactive Management for providing information on the methodologies being used by their professionals and/or their clients.

Special thanks to Messrs. Charles Alleyne and Edwin Twum-Danzo and Misses Denise Caesar, Audrey Gates, and Serita Sanders [Graduate Students in the School of Engineering, Howard University] for their enthusiastic support with the acquisition and review of documents and compilation of the background information.

EXECUTIVE SUMMARY

The purpose of this research effort is to study systems engineering methodologies in light of changing environments and changing needs. The results of this investigation will be used to identify and validate new methodologies with potential applications to the National Aeronautics and Space Administration's (NASA's) systems life-cycle planning processes.

The study has been conducted in two phases. Phase One was a study of NASA's systems projects, its organization, resources, and environment to identify uniqueness, factors that affect the successful application of systems engineering methodologies to NASA's projects, and the need for new methodologies. Phase Two involved evaluating methodologies, tools, and techniques with potential for application to NASA's systems projects, and making recommendations to NASA.

Within each phase of the project, the technical approach involved the following major steps: identifying sources of information; conducting library searches; reviewing documents; conducting interviews; analyzing data; and documenting the results. An additional aspect of the program included extending invitations to experts in the field of systems engineering to make presentations on selected topics at Howard University or at NASA's Goddard Space Flight Center (GSFC).

The Phase One study concluded that the Systems Management Policy (SMP), the primary methodology being used by the Mission Operations and Data Systems Directorate and its subordinate--the Networks Division (ND)--the sponsor of this effort, is considered to be very effective by its users; and as such, many of the needs for systems engineering methodologies are currently being satisfied. The study identified, however, some un-met needs or areas of weakness in either the methodology or the manner in which it is applied, as follows:

(1) Deficiencies in the Systems Engineering Methodology being used by the ND

- While the SMP suggests that it can be tailored to meet the needs of different projects (types and sizes), in practice, considerable effort is required to streamline it for the very small projects, and the feasibility of such streamlining has been questioned by some project managers.
- The details (steps, tasks, activities) of the methodology are not clearly defined.

- While the methodology specifies the type of documents that should be produced at different points in the system development cycle, it does not provide sufficient details about the content and structure of those documents.
- The methodology provides no assistance with projecting or predicting future requirements.
- The methodology does not respond adequately to changing and emerging requirements. Thus, at the time the system is implemented it is usually responding to requirements of several years earlier, not the current requirements.
- The methodology does not support the development of systems in situations where it is not possible to define the requirements [institutional systems or systems on the cutting edge of technology].
- The methodology provides minimal support with tools and techniques for communicating among participants on a major project, e.g., graphics and prototyping.
- The methodology is not sufficiently flexible with regard to scheduling of tasks to accommodate changing priorities and funding levels and to maximize the effective utilization of resources.
- The methodology does not address adequately the possibility of extensively redesigning or modifying an existing system to incorporate new requirements and capabilities and extend its useful life, as opposed to retiring that system and developing a completely new system to replace it. Modifying an existing system tends to shorten the time to have a capability in place.

(2) Problems with the Application and Management of Systems Engineering within the ND

- While the ND and the MO&DSD frequently work with other directorates and major organizational units within NASA on large agency-wide projects, each organizational unit uses its own methodology, with the project manager having responsibility for coordinating and negotiating approaches.
- While proposals are reviewed extensively for adherence to the requirements of the RFP, some staff members are concerned that methodology is not given adequate importance among the evaluation criteria and in the review and evaluation process.

- Systems engineering support contractors are generally involved in routine systems analysis work, and they are usually not used effectively for systems engineering management or in supporting the application of systems engineering methodologies to major ND projects.
- While project management plans are reviewed administratively, a concern of some staff members is that they are not reviewed rigorously from a systems engineering perspective.
- Some of the Division's smaller projects, the sustaining engineering projects, are not developed with a formal systems engineering methodology.

The primary focus of this phase (Phase Two) has been to evaluate available methodologies, using evaluation criteria developed in Phase One. In this regard, six methodologies from government and industry have been evaluated together with the SMP. The conclusions and recommendations are summarized as follow.

While each of the methodologies reviewed has some unique strengths, no individual methodology will satisfy all of the needs of the ND. Thus, the process of developing methodologies that are more suitable for the ND should involve extracting desired features from a variety of sources, integrating, and testing to verify that they will satisfy the requirements. The methodologies reviewed can improve on the methodology in use by the ND in the following areas:

- partitioning the project into phases, work breakdown items, tasks, functions, and work packages which can be used in planning the project;
- handling new information, feedback, or unforeseen circumstances;
- acquiring the system or its components by procurement;
- scheduling of project resources;
- identifying and selecting human and material resources;
- specifying the contents and structure of documents needed at different points in the development process;
- management procedures to ensure that the methodology is being applied as intended;
- ways of addressing critical considerations such as national security, and risk to humans and the environment;
- clearer, more precise, and more complete documentation;

- techniques for retaining key personnel on the project; and
- tools and techniques for redesigning and modifying to add capabilities and extend the useful life of the system.

Needs of the ND that cannot be satisfied by the methodologies reviewed include:

- improving the process of scoping the basic methodology to projects of different types and sizes;
- tools and techniques for predicting or projecting future requirements;
- strategies and techniques for designing and developing systems in the absence of specific requirements; and
- tools and techniques, including graphics and prototyping, for communicating among individuals and organizations working on major systems projects.

This study recommends the following:

- (1) That the ND develop a statement of policy and related design principles to guide the project manager, systems engineering manager, and staff in areas where the methodology fails to provide adequate guidance.
- (2) That the ND consider training in project management and systems engineering to be an on-going requirement for successful management of large systems development projects.
- (3) That the ND resolve the weaknesses in the SMP by incorporating desirable features from the other methodologies reviewed in this effort and from other sources. Table 13-1 shows the methodologies that are judged to be superior to the SMP in the different problem areas.
- (4) That the ND incorporate more participative design, prototyping, and consensus building techniques in its systems development process to provide some relief in the difficult and yet unresolved problem areas of communication among stakeholders (design team: contractors and staff, users, managers, etc.), and definition, projection/prediction of requirements.

TABLE OF CONTENTS

	Page
PREFACE	1
ACKNOWLEDGEMENT	ii
EXECUTIVE SUMMARY	iii
TABLE OF CONTENTS	vii
LIST OF FIGURES	x
LIST OF ABBREVIATIONS AND ACRONYMS	xi
 1. INTRODUCTION	 1-1
1.1 ORGANIZATION OF THIS REPORT	1-2
REFERENCES	1-4
 2. BACKGROUND	 2-1
2.1 DEFINITIONS	2-1
2.2 OBJECTIVES OF THIS PHASE	2-3
REFERENCES	2-4
 3. APPROACH	 3-1
 4. SYSTEM DEVELOPMENT PROBLEMS: THE NEED FOR SYSTEM DEVELOPMENT METHODOLOGIES	 4-1
4.1 NEED OF THE ND FOR SYSTEMS DEVELOPMENT METHODOLOGIES .	4-1
4.2 NEEDS IDENTIFIED BY GOVERNMENT AGENCIES AND INDUSTRY FOR SYSTEMS DEVELOPMENT METHODOLOGIES	4-2
4.3 SUMMARY	4-5
REFERENCES	4-6
 5. FRAMEWORK FOR EVALUATING METHODOLOGIES	 5-1
5.1 BASIC APPROACHES TO SYSTEMS DEVELOPMENT.....	5-1
5.2 EVOLVING APPROACHES TO SYSTEM DEVELOPMENT	5-3
5.3 CRITERIA FOR EVALUATING METHODOLOGIES	5-7
5.4 SUMMARY	5-12
REFERENCES	5-13
 6. REVIEW OF THE SYSTEMS MANAGEMENT POLICY OF THE MISSION OPERATIONS AND DATA SYSTEMS DIRECTORATE.....	 6-1
6.1 THE WORK BREAKDOWN STRUCTURE	6-1
6.2 THE SYSTEM LIFE CYCLE MODEL	6-8
6.3 SUMMARY	6-14
REFERENCE	6-14
 7. REVIEW OF HUGHES SYSTEMS ENGINEERING METHODOLOGY	 7-1
7.1 CONCEPT DEFINITION PHASE	7-2
7.2 SYSTEM DESIGN	7-3
7.3 DETAILED DESIGN	7-4
7.4 SYSTEM INTEGRATION AND TEST	7-5

TABLE OF CONTENTS CONT'D

	Page
7.5 PRODUCTION	7-6
7.6 SUMMARY	7-7
REFERENCE	7-7
 8. REVIEW OF COMPUTER SCIENCES CORPORATION DIGITAL SYSTEMS DEVELOPMENT METHODOLOGY	 8-1
8.1 REQUIREMENTS DEFINITION	8-2
8.2 DESIGN	8-2
8.3 IMPLEMENTATION	8-4
8.4 INTEGRATION AND TEST	8-5
8.5 TURNOVER	8-5
8.6 PROJECT MANAGEMENT	8-6
8.7 SYSTEM ENGINEERING	8-7
8.8 SUMMARY	8-8
REFERENCE	8-8
 9. REVIEW OF SYSTEMS ENGINEERING FOR INTEGRATED HARDWARE/SOFTWARE APPLICATIONS	 9-1
9.1 CONCEPTUAL DESIGN	9-1
9.2 SYSTEM REQUIREMENTS ANALYSIS	9-2
9.3 SYSTEM DESIGN	9-3
9.4 SUBSYSTEM IMPLEMENTATION	9-5
9.5 SYSTEM INTEGRATION, VALIDATION, DELIVERY, O AND M AND DISPOSAL	9-6
9.6 GENERIC LIFE CYCLE ACTIVITIES	9-8
9.7 SUMMARY	9-8
REFERENCE	9-9
 10. REVIEW OF JET PROPULSION LABORATORY SYSTEM DEVELOPMENT METHODOLOGY	 10-1
10.1 JPL's PRINCIPLE OF SYSTEM DEVELOPMENT	10-1
10.2 SYSTEM DEVELOPMENT SPECIFICATION	10-3
10.3 SYSTEM DEVELOPMENT EXECUTION	10-6
10.4 SUMMARY	10-7
REFERENCES	10-9
 11. REVIEW OF DEFENSE SYSTEMS MANAGEMENT COLLEGE SYSTEMS ENGINEERING MANAGEMENT METHODOLOGY	 11-1
11.1 THE LIFE CYCLE PHASES	11-1
11.2 THE SYSTEMS ENGINEERING PROCESS	11-5
11.3 SUMMARY	11-7
REFERENCE	11-8
 12. REVIEW OF SYSTEM ENGINEERING AND ANALYSIS	 12-1
12.1 SYSTEMS ENGINEERING FUNCTIONS	12-1

TABLE OF CONTENTS CONT'D

	Page
12.2 SYSTEMS ENGINEERING MANAGEMENT	12-6
12.3 SUMMARY	12-7
REFERENCE	12-7
13. EVALUATING SELECTED METHODOLOGIES	13-1
13.1 SCOPE AND STRUCTURE OF THE METHODOLOGY	13-1
13.2 FLEXIBILITY	13-1
13.3 ACCOUNTABILITY	13-4
13.4 DOCUMENTATION OF THE METHODOLOGY	13-5
13.5 SPECIAL CONSIDERATIONS OF THE ND	13-6
13.6 ANALYSIS	13-8
13.7 SUMMARY	13-9
14. TOOLS AND TECHNIQUES	14-1
14.1 PARTICIPATIVE DESIGN	14-1
14.2 PROTOTYPING	14-2
14.3 CONSENSUS METHODOLOGIES	14-4
14.4 PROJECT FORCE FIELD ANALYSIS	14-9
14.5 SUMMARY	14-11
REFERENCES	14-12
15. ORGANIZATIONAL CULTURE AND PHILOSOPHY	15-1
15.1 WHAT IS MOST IMPORTANT	15-1
15.2 THE ROLE OF MANAGEMENT	15-1
15.3 THE ROLE OF THE PROJECT MANAGER	15-2
15.4 THE ROLE OF THE STAFF	15-2
15.5 DESIGN PRINCIPLES	15-3
15.6 SUMMARY	15-6
REFERENCES	15-7
16. CONCLUSIONS AND RECOMMENDATIONS	16-1
16.1 CONCLUSIONS	16-2
16.2 RECOMMENDATIONS	16-5

LIST OF FIGURES

	Page
5-1: TRADITIONAL APPROACH TO SYSTEM DEVELOPMENT	5-1
5-2: PROTOTYPING APPROACH TO SYSTEM DEVELOPMENT	5-2
5-3: INCREMENTAL APPROACH TO SYSTEM DEVELOPMENT	5-3
5-4: PRODUCT, PROCESS, AND SUPPORT LIFE CYCLES	5-5
5-5: LIFE CYCLE DESIGN ACTIVITIES AND INTERACTIONS	5-6
5-6: THREE FUNDAMENTAL LEVELS OF SYSTEMS ENGINEERING	5-8
5-7: FRAMEWORK FOR SYSTEMS ENGINEERING	5-9
6-1: WORK BREAKDOWN STRUCTURE FOR A TYPICAL MO&DSD PROJECT	6-2
6-2: ELEVEN PHASES OF THE SYSTEM DEVELOPMENT LIFE CYCLE ...	6-9
6-3: DOCUMENTATION REQUIREMENTS FOR THE LIFE CYCLE	6-10
7-1: PHASES OF THE HUGHES SYSTEMS DEVELOPMENT PROCESS	7-1
7-2: SYSTEMS ENGINEERING AS A CONTINUAL PROCESS	7-2
7-3: THE DESIGN PROCESS: ALLOCATE, INTEGRATE, VALIDATE	7-5
7-4: THE DETAIL DESIGN PATH	7-6
8-1: SUMMARY OF PLANS, PRODUCTS, REVIEWS, AND BASELINES FOR DSDM	8-1
8-2: THE REQUIREMENTS DEFINITION PROCESS	8-3
8-3: FUNCTIONS AND ACTIVITIES OF THE PROJECT MANAGER	8-6
10-1: THE D-5000 MODEL ARCHITECTURE	10-4
10-2: SYSTEM DEVELOPMENT PHASES AT SUCCESSIVE SYSTEM LEVELS	10-8
11-1: DoD LIFE CYCLE PHASES AND MILESTONES	11-2
11-2: DSMC SYSTEMS ENGINEERING PROCESS	11-6
12-1: FEEDBACK IN THE SYSTEMS ENGINEERING PROCESS	12-2
14-1: THE PROTOTYPING PROCESS	14-3
14-2: IDEAWRITING	14-5
14-3: NOMINAL GROUP TECHNIQUE	14-7
14-4: INTERPRETIVE STRUCTURAL MODELING	14-8
14-5: OPTIONS FIELD METHOD	14-10
14-6: FORCE FIELD ANALYSIS STRUCTURE	14-11

LIST OF ABBREVIATIONS AND ACRONYMS

B&F	- Blanchard and Fabrycky
BCWP	- Budgeted Cost of Work Performed
CalTech	- California Institute of Technology
CA	- Configuration Audit
CCB	- Configuration Control Board
CCR	- Configuration Change Request
CDR	- Critical Design Review
CI	- Configuration Item
CIM	- Center for Interactive Management
CM	- Configuration Management
CSC	- Computer Sciences Corporation
DAA	- Designated Approval Authority
DoD	- Department of Defense
DSDM	- Digital Systems Development Methodology
DSMC	- Defense Systems Management College
EC	- Engineering Change
FCA	- Functional Configuration Audit
FSD	- Full Scale Development
GDT	- Generic Design Theory
GSFC	- Goddard Space Flight Center
HBCU	- Historically Black Colleges and Universities
HUGHES	- HUGHES Aircraft Company
ICS	- Integrated Computer Systems
ILSP	- Integrated Logistic Support Plan
IM	- Interactive Management
JMSNS	- Justification for Major System New Start
JPL	- Jet Propulsion Laboratory
JSC	- Johnson Space Center
MO&DSD	- Mission Operations and Data Systems Directorate
MSFC	- Marshall Space Flight Center
NASA	- National Aeronautics and Space Administration
ND	- Networks Division
NGT	- Nominal Group Technique
NSF	- National Science Foundation
NTIS	- National Technical Information System
O&M	- Operations and Maintenance
PCA	- Physical Configuration Audit
PDR	- Preliminary Design Review
PMP	- Project Management Plan
PPP	- Phased Project Planning
QA	- Quality Assurance
RFP	- Request for Proposal
SDR	- System Design Review
SDS	- System Development Specification
SEMP	- Systems Engineering Management Plan
SMP	- System Management Policy
SORD	- Systems and Operations Requirements Document
SSORCE	- Systems, Software, and Operations Resource Center

SSR	- System Specification Review
SRR	- System Requirements Review
TEMP	- Test and Evaluation Master Plan
US	- United States
WBS	- Work Breakdown Structure

1. INTRODUCTION

The purpose of this effort is to study systems engineering methodologies in light of changing environments and changing needs. The results of this investigation will be used to identify and validate new methodologies with potential applications to the National Aeronautics and Space Administration's (NASA's) systems life-cycle planning processes.

The study has been conducted in two phases. Phase One was a study of NASA's systems projects, its organization, resources, and environment to identify uniqueness, factors that affect the successful application of systems engineering methodologies to NASA's projects, and the need for new methodologies. Phase Two involves evaluating methodologies, tools, and techniques with potential for application to NASA's systems projects, and making recommendations to NASA.

Within each phase of the project, the technical approach involved the following major steps: identifying sources of information; conducting library searches; reviewing documents; conducting interviews; analyzing data; and documenting the results. An additional aspect of the program included extending invitations to experts in the field of systems engineering to make presentations on selected topics at Howard University or at NASA's Goddard Space Flight Center (GSFC).

Surveys of NASA's personnel and contractors, conducted as part of Phase One of this effort, indicated that the System Management Policy (SMP), which describes the systems engineering Methodology of the Mission Operations and Data Systems Directorate (MO&DSD), had been well received by its users in the Networks Division (ND); that some aspects of the methodology are being incorporated into the Statement of Work and Specifications sections of the Request for Proposal (RFP), which eventually become parts of the contractual requirements; and that most contractors who regularly support the ND are aware of the SMP, and voluntarily propose methodologies and approaches that are consistent with it.

The results of Phase One have been presented to MO&DSD [1], at an annual conference sponsored jointly by NASA and the Historically Black Colleges and Universities, [2,3] and at the annual conference of the National Technical Association. [4]

A major product of Phase One was a set of criteria for evaluating methodologies. These criteria have been used in this phase to evaluate six methodologies from government and industry for applicability to the ND projects.

The study concludes that none of the methodologies reviewed by itself will adequately address all of the ND's methodological

needs; extracting some desirable features from all of these methodologies will satisfy a subset of the un-met needs; and other sources and strategies will be required to address the still unsatisfied needs.

The study recommends that the ND improve its systems engineering methodology by incorporating desirable features from the methodologies reviewed. Table 13-1 indicates which methodologies are judged to be more effective than the SMP for the different factors included in the evaluation criteria. The study recommends participative design, prototyping, and consensus building techniques as ways to address some of the still unresolved problems of inadequate communication among the participants or stakeholders, and defining and projecting/predicting requirements.

1.1 ORGANIZATION OF THIS REPORT

This report is written in sixteen chapters, as summarized below:

- Chapter One: **"INTRODUCTION"** contains a general introduction to the study and this description of the organization of the Phase Two Report.
- Chapter Two: **"BACKGROUND"** contains the definitions, background information on systems engineering, the objectives, and scope of the study.
- Chapter Three: **"APPROACH"** describes the stepwise methodology used in this study.
- Chapter Four: **"SYSTEM DEVELOPMENT PROBLEMS: THE NEED FOR SYSTEM DEVELOPMENT METHODOLOGIES"** reviews the ND's needs for systems engineering methodologies identified in Phase One of this project, and provides background information on the types of problems encountered in other government agencies and in industry.
- Chapter Five: **"FRAMEWORK FOR EVALUATING METHODOLOGIES"** describes some of the basic approaches to system development, and discusses the organization and structure of the evaluation criteria used in this phase to evaluate methodologies for applicability to the ND' systems development projects.
- Chapter Six: **"REVIEW OF THE SYSTEMS MANAGEMENT POLICY OF THE MISSION OPERATIONS AND DATA SYSTEMS DIRECTORATE"** summarizes the Systems Engineering Methodology being used by the ND.

- Chapter Seven: **"REVIEW OF HUGHES SYSTEMS ENGINEERING METHODOLOGY"** summarizes the systems engineering methodology being used by HUGHES Aircraft Company.
- Chapter Eight: **"REVIEW OF COMPUTER SCIENCES CORPORATION DIGITAL SYSTEMS DEVELOPMENT METHODOLOGY"** summarizes the systems engineering methodology being used by Computer Sciences Corporation.
- Chapter Nine: **"REVIEW OF SYSTEMS ENGINEERING FOR INTEGRATED HARDWARE/SOFTWARE APPLICATIONS"** provides a summary of a training program in systems engineering being delivered commercially by Learning Tree International, formerly, Integrated Computer Systems.
- Chapter Ten: **"REVIEW OF JET PROPULSION LABORATORY SYSTEMS DEVELOPMENT METHODOLOGY"** summarizes the systems engineering methodology being developed by the Jet Propulsion Laboratory.
- Chapter Eleven: **"REVIEW OF DEFENSE SYSTEMS MANAGEMENT COLLEGE SYSTEMS ENGINEERING MANAGEMENT METHODOLOGY"** summarizes the systems engineering methodology being used by the Department of Defense as part of its acquisition program.
- Chapter Twelve: **"REVIEW OF SYSTEMS ENGINEERING AND ANALYSIS"** provides a summary of a textbook on systems engineering by Blanchard and Fabrycky.
- Chapter Thirteen: **"EVALUATING SELECTED METHODOLOGIES"** discusses the application of the evaluation criteria to the selected methodologies.
- Chapter Fourteen: **"TOOLS AND TECHNIQUES"** describes tools and techniques with potential for application to the ND's system development project.
- Chapter Fifteen: **"ORGANIZATIONAL CULTURE AND DESIGN PHILOSOPHY"** discusses the need to develop a systems development culture or philosophy to guide systems development activities in areas where the methodology is vague or inadequate.
- Chapter Sixteen: **"CONCLUSIONS AND RECOMMENDATIONS"** presents the conclusions and recommendations of Phase Two of this effort, and Appendix A shows a list of Phase Two speakers.

REFERENCES

1. Paul, Arthur S., et al, A Dynamic Systems Engineering Methodology Research Study, (Phase One: Understanding NASA's Systems Projects and the Need for Systems Engineering Methodologies), Howard University, November 1988.
2. Alleyne, Charles, et al, A Dynamic Systems Engineering Methodology Research Study, (Phase One: Understanding NASA's Systems Projects and the Need for Systems Engineering Methodologies)", Paper presented at the Joint NASA/HBCU Conference, May 1989.
3. Gates, Audrey, et al, "Suggested Criteria for Evaluating Systems Engineering Methodologies", Paper Presented at the Joint NASA/HBCU Conference, May 1989.
4. Caesar, Denise and Arthur S. Paul, "Managing Systems Engineering Activities", Paper Presented at the National Technical Association Conference, July 1989.

2. BACKGROUND

The earliest use of a planning methodology, agency-wide, at NASA was the introduction of Phased Project Planning (PPP) in October 1965. This approach consisted of four phases as follows: advanced studies (Phase A), project definition (Phase B), design (Phase C), and development/operation (Phase D).[1, p. 161] In June 1982, the Network Control Center Division published its Project Management Plan (PMP).[2] The PMP was supplemented by a Software Management Plan published in March 1984. The stated purpose of the Software Management Plan was to strengthen the management of the software development process, verification, and implementation.[3] In October 1986, NASA introduced at the Headquarters a methodology for software acquisition to be used throughout the Agency.[4] In the same year the MO&DSD introduced the SMP, as a methodology for systems development that was to be used for all projects being implemented within the Directorate.[5]

2.1 DEFINITIONS

Because systems engineering is a relatively new field of study and because of diversity in the definition of "system", various definitions and concepts of systems engineering are discussed in this section.

2.1.1 Definitions of Systems Engineering

A widely used definition of systems engineering from the Department of Defense is: "Systems Engineering is the application of scientific and engineering efforts to (a) transform an operational need into a description of system performance parameters and a system configuration through the use of an iterative process of definition, synthesis, analysis, design, test, and evaluation; (b) integrate related technical parameters and ensure compatibility of all physical, functional, and program interfaces in a manner that optimizes the total system definition and design; (c) integrate reliability, maintainability, safety, survivability, human, and other such factors into the total engineering effort to meet cost, schedule, and technical performance objectives."[6]

The Defense Systems Management College describes systems engineering as follows: "In its simplest terms, systems engineering is both a technical process and a management process. To successfully complete the development of a system, both aspects must be applied throughout the system life-cycle. A system life cycle begins with the user's needs, which are expressed as constraints, and the capability requirements needed to satisfy mission objectives. Systems engineering is essential in the

earliest planning period, in conceiving the system concept, and defining requirements for the system. As the detailed design is being done, systems engineers assure balanced influence of all required design specialties, resolve interface problems, perform design reviews, perform trade-off analyses, and assist in verifying performance. During the Production phase, systems engineering is concerned with verifying system capability and maintaining the system baseline, and forms an analytical framework for producibility analysis. During the Operation and Support phase, systems engineering evaluates proposed changes to the systems, establishes their effectiveness, and facilitates the effective incorporation of changes, modifications, and updates."[7, P. 1-3]

Sage presents systems engineering from a managerial standpoint as: "systems engineering is management technology to assist and support policy making, planning, decision making, and associated resource allocation or action deployment. It accomplishes this by quantitative and qualitative formulation, analysis, and interpretation of the impacts of action alternatives upon the need perspectives, the institutional perspectives, and the value perspectives of clients to a system engineering study."[8]

The definition proposed in Phase One of this effort is: "Systems Engineering is the application of mathematical and scientific principles to practical ends, in the life-cycle of a system."[9] This definition suggests that there is a role for systems engineering beyond the systems development phase and beyond simply developing plans for the operations phase. Systems engineering management should extend through the operations, maintenance, modification, and retirement activities of large complex systems.

2.1.2 Other Definitions and Concepts

The following definitions will help to present and/or clarify some of the key terms and concepts in the field of systems engineering and systems planning.

A System is an interconnection of parts or components to form a unity or organic whole to achieve some specified objectives.

Engineering is the application of mathematical and scientific principles to practical ends, as the design, construction, and operation of economical and efficient structures, equipment, and systems.

A Methodology for Systems Engineering is a carefully developed, relatively complex procedure or process for applying mathematical and scientific principles during the life-cycle of a system.

Requirements: condition and capability needed by user to solve a problem or achieve an objective. System requirements specify:

- Function: what it does,
- Performance: how well it does it,
- Environment: under what conditions,
- Interfaces: inter- and intra- system relationships,
- Testability: ease and thoroughness of validation,
- Reliability: success probability and fault tolerance,
- Maintainability: ease of service,
- Operability: ease of operation, and
- Safety: protection of itself and "others"; and
- System requirements also typically (but not always) affect more than one subsystem.[10]

Attributes of good requirements:

- Identified and acknowledged: every requirement identified and accepted separately;
- Unambiguous: Limited to a single interpretation;
- Clear: Eschews obfuscation;
- Concise: No rambling verbiage or unnecessary language;
- Implementation-free: Focus on what not how;
- Testable: Can be verified by a finite, cost-effective process;
- Traceable: Hierarchical correlation;
- Complete: Taken together, they specify the entire job; and
- Feasible: Possible to construct this system within the available resources (schedule, cost, staff).[10]

Specifications are documents that clearly and accurately describe requirements for systems, items, materials, or services including methods by which it will be determined that requirements have been met. Specifications provide a basis for: documenting requirements, controlling incremental development, and providing visibility in development process. Types of specifications include:

- System/segment specification,
- Development specification,
- Product specification,
- Process specification, and
- Material specification.[11, P. 2-3]

Some aspects of systems engineering are covered in the general literature under related fields, such as: Decision Making, Systems Acquisition, Systems Analysis, Systems Planning, and Systems Management.

2.2 OBJECTIVES OF THIS PHASE

The objectives developed for this phase of the study include:

- to identify existing methodologies with potential for application to NASA's systems projects;
- to evaluate existing methodologies (including NASA's), based on the criteria developed in Phase One;
- conditioned on the outcome of this evaluation, to synthesize new methodologies for application to NASA's systems projects; and
- to make recommendations on systems engineering methodology to NASA.

REFERENCES

1. NASA, Managing NASA in the Apollo Era, Washington, D.C., 1982.
2. Networks Control Center Division, Network Control Center Project Management Plan, STDN No. 907, June 1982.
3. Networks Control Center Division, Network Control Center Data System Software Management Plan, STDN No. 907.1, March 1984.
4. Office of Safety, Reliability, Maintainability, and Quality Assurance, NASA Software Acquisition Life Cycle Chart, October 1986.
5. MO&DSD, System Management Policy, 1986.
6. Military Standard: Specification Practices, MIL-STD-490A, Department of Defense, June 1985.
7. Systems Engineering Management Guide, Second Edition, (Fort Belvoir, Virginia: Defense Systems Management College), December 1986.
8. Sage, Andrew P., "Systems Engineering", (Notes from a presentation made at Goddard), February 6, 1989.
9. Paul, Arthur S., et al, A Dynamic Systems Engineering Methodology Research Study, (Phase One: Understanding NASA's Systems Projects and the Need for Systems Engineering Methodologies), Howard University, November 1988.
10. The Learning Tree, Course 348: Systems Engineering for Integrated Hardware/Software Applications, (Vienna,

Virginia: Integrated Computer Systems Publishing Company, Inc.), 1988.

11. Systems Engineering Seminars for General Motors, HUGHES Aircraft Company, 1987.
12. Hall, Arthur D., A Methodology for Systems Engineering, (New York: Van Nostrand Reinhold Company), 1962.
13. Sage, Andrew P., Methodology for Large-Scale Systems, (New York: McGraw-Hill Book Company), 1977.
14. Coleman, D., Parker, N., and Scipio, A., Introduction to Urban Systems Engineering, (Monograph), 1975.
15. Blanchard, B. S. and Fabrycky, W. J., Systems Engineering and Analysis, (New Jersey: Prentice-Hall), 1981.
16. Warfield, J. N. and Hill, J. D., An Assault on Complexity, (A Battelle Monograph Number 3), April 1973.
17. Warfield, J. N. and Hill, J. D., A Unified Systems Engineering Concept, (Battelle Memorial Institute), 1972.
18. Churchman, C. West, The Systems Approach, (New York: Dell Publishing Co. Inc.), 1968.
19. Hawthorne, R.W., Farris, D.R., Sage, A.P., "Application of Unified Program Planning to Public and Societal Systems: A Case Study", Portraits of Complexity: Applications of Systems Methodologies to Societal Problems, A Battelle Monograph, Number 9, June 1975.

3. APPROACH

This phase of the project involved evaluating methodologies, tools, and techniques with potential for application to NASA's systems projects, and making recommendations to NASA. The approach used in this phase involves the following tasks:

- Kick Off Meeting with NASA,
- Identify Sources of Information,
- Develop Data Collection Instruments,
- Conduct Library Searches,
- Review Documents,
- Analyze Data, and
- Document Results.

A Kick Off Meeting was conducted with personnel from the Networks Division at GSFC. The purposes of the meeting were to review the results of Phase One and to introduce new members of the Project Team to the Technical Officer assigned to this project and NASA's management personnel. During that meeting, the Project Team presented the goals and objectives and an outline of the approach to Phase Two.

Through brainstorming and discussions with NASA's technical staff, it was determined that most of the data for this project would come from companies involved in developing major systems for NASA and DoD, from government agencies with responsibility for similar systems, and from the general literature on systems engineering.

Two data collection instruments were developed: one for extracting information from complete methodologies provided by contractors and government agencies and the other for extracting information on tools and techniques and other items of importance to the project from journal articles, textbooks, and notes from presenters.

Library searches were conducted at: NASA's Scientific and Technical Library; the library at Goddard Space Flight Center; the Library of Congress; the National Technical Information Services (NTIS); and Howard University Libraries. These searches involved using key words selected during a working session of the Project Team. The searches were conducted in an iterative process. The results of each search were reviewed, and the Project Team decided, as a group, whether to continue, using a different combination of key words, or to terminate the process.

The title was used as a basis for selecting the documents to be reviewed. Some documents were reviewed at the library, but the more important ones were borrowed, and reviewed thoroughly at the project office. Excerpts were extracted, and summaries prepared,

as necessary, for the data analysis and preparation of the final report.

In analyzing the data, each methodology was reviewed to determine the extent to which it incorporated the features that were pre-selected as the evaluation criteria. The results were tabulated for easy comparison, and documented in this report of the activities and results of Phase Two.

4. SYSTEM DEVELOPMENT PROBLEMS: THE NEED FOR SYSTEM DEVELOPMENT METHODOLOGIES

This chapter summarizes the ND's needs for systems engineering methodologies, as identified in Phase One of this project. It then reviews the general literature to determine similarities and differences between the problems of the ND and those of other government agencies and private enterprises. The results of the comparison is presented in a summary at the end of the chapter.

4.1 NEEDS OF THE ND FOR SYSTEMS DEVELOPMENT METHODOLOGIES

The study conducted in Phase One, which involved surveying program managers and systems engineers within the ND and their contractors, concluded that the SMP, the primary methodology being used by the MO&DSD and its subordinate--the ND--is considered to be very effective by its users. Thus, many of the needs for systems engineering methodologies are currently being met. The study identified, however, some un-met needs or areas of weakness in either the methodology or the manner in which it is applied, as follows:

- While the ND and the MO&DSD frequently work with other directorates and major organizational units within NASA on large agency-wide projects, each organizational unit uses its own methodology, with the project manager having responsibility for coordinating and negotiating approaches.
- While proposals are reviewed extensively for adherence to the requirements of the RFP, some staff members are concerned that methodology is not given adequate importance among the evaluation criteria and in the review and evaluation process.
- Systems engineering support contractors are generally involved in routine systems analysis work, and they are usually not used effectively for systems engineering management or in supporting the application of systems engineering methodologies to major ND projects.
- While the SMP suggests that it can be tailored to meet the needs of different projects (types and sizes), in practice, considerable effort is required to streamline it for the very small projects, and the feasibility of such streamlining has been questioned by some project managers.
- While project management plans are reviewed administratively, a concern of some staff members is that they are not reviewed rigorously from a systems engineering perspective.

- The details (steps, tasks, activities) of the methodology are not clearly defined.
- While the methodology specifies the type of documents that should be produced at different points in the system development cycle, it does not provide sufficient details about the content and structure of those documents.
- The methodology provides no assistance with projecting or predicting future requirements.
- The methodology does not respond adequately to changing and emerging requirements. Thus, at the time the system is implemented it is usually responding to requirements of several years earlier, not the current requirements.
- The methodology does not support the development of systems in situations where it is not possible to define the requirements [institutional systems or systems on the cutting edge of technology].
- The methodology provides minimal support with tools and techniques for communicating among participants on a major project, e.g., graphics and prototyping.
- The methodology is not sufficiently flexible with regard to scheduling of tasks to accommodate changing priorities and funding levels and to maximize the effective utilization of resources.
- The methodology does not address adequately the possibility of extensively redesigning or modifying an existing system to incorporate new requirements and capabilities and extend its useful life, as opposed to retiring that system and developing a completely new system to replace it. Modifying an existing system tends to shorten the time to have a capability in place.
- Some of the Divisions smaller projects, the sustaining engineering projects are not developed with a formal systems engineering methodology.

4.2 NEEDS IDENTIFIED BY GOVERNMENT AGENCIES AND INDUSTRY FOR SYSTEMS DEVELOPMENT METHODOLOGIES

Some of the more basic need for systems development methodologies are summarized from Nadler [1, PP. 84-86] as:

- to improve an existing system or product,
- to diagnose or remedy trouble,
- to develop new system or product,

- to develop a new use for an existing system, and
- to address low productivity and poor management in industry.

Dos Santos [2, P. 35] claims that today's systems are:

- Unstructured,
- Span departmental boundaries, and
- Take many years to complete.

Sage [3] discusses some of the problems associated with developing large scale systems as follows: "In reality, there are many difficulties associated with the production of functional, reliable, and trustworthy systems of large scale and scope. There are many studies which indicate that:

- Large systems are expensive;
- System capability is often less than promised and expected;
- System deliveries are often quite late;
- Large system cost over-runs often occur;
- Large system maintenance is complex and error prone;
- Large system documentation is inappropriate and inadequate;
- Large systems are often cumbersome to use and system design for human interaction is generally lacking;
- Individual subsystems often cannot be integrated;
- Large systems often cannot be transitioned to a new environment or modified to meet evolving needs;
- Large system performance is often unreliable;
- Large systems often do not perform according to specifications; and
- System requirements often do not adequately capture user needs."

The Computer Sciences Corporation [4] summarizes the general problems with systems development methodologies as follows:

- Difficulty in measuring the true status of the development effort accurately, especially when software is a principal element of the system;
- Implementation of design whose poor quality does not surface until the finished product is either subjected to final testing or installed for operation; and
- High life-cycle costs resulting from a system that was not designed for reusability or maintainability.

Mumford and others [5] identify the problems of current design approaches as:

- The need of users to assume a new and unfamiliar role,
- Difficulties communicating with colleagues, and

- Differences in expectations between design group members and management.

Lackman [6] identifies the two major problems of systems development as:

- Inability to maintain (control) a delivery schedule and
- Inability to manage their client.

During August and September 1988, Government and Industry Program Managers from 22 Smart Munitions Programs participated in four workshops. The purpose of which was to identify critical factors which inhibit their ability to meet cost and schedule objectives and to propose solutions to those inhibiting factors.[7]

The workshops were conducted as a collaborative effort between the Defense Systems Management College (DSMC) and the Center for Interactive Management (CIM) at George Mason University. The study team used the Interactive Management System, a set of computer-assisted tools for allowing groups to find and define problems, design alternatives for addressing these problems, and select the preferred alternative(s). These tools are discussed further in Chapter 14.

At the conclusion of the workshops, a total of 295 critical inhibitors and 274 solution ideas were identified. A listing of the more important issues follows:

A. Program Manager's Responsibility and Authority:

- Dilution of program manager's authority,
- Too many inhibitors outside the control of the program manager,
- Government program managers cannot control programs,
- Lack of adequate program management staff and motivating factors to maintain,
- Micro-management at all levels of oversight, and
- Lack of program manager's flexibility (ability) to deal with change.

B. Budgetary Limitations and Fluctuations:

- Year-to-year instabilities in budget and procurement quantities,
- Lack of program funding stability,
- Lack of consistent budget for planning purposes, and
- Annual production budget fluctuations leading to bath tubs and gaps.

C. Requirements Specification:

- Changing requirements,

- Inability to lock in requirements,
- Mis-estimation of technical difficulties,
- Changes in policy and specifications, and
- Poorly defined and changing technical performance requirements.

D. Other:

- Unrealistic program plans/schedules and associated funding profiles,
- Lack of adequate engineering disciplines during all phases of the acquisition process,
- Lack of regulation and historical approaches to cleansing,
- Existence of extensive special interest bureaucracy within the acquisition infrastructure,
- Instability of Department of Defense (DoD) and Congressional support for programs,
- Illogical competition, and
- Changing interpretation of criteria for successful demonstration of requirements.

4.3 SUMMARY

While there are many perspectives on the nature and scope of the problem and the needs for systems development methodologies, some common baselines are:

- today's large systems are complex and resource intensive [from both development and operational standpoints], and, under ideal conditions, their development can be time consuming and cross organizational and business boundaries;
- current methodologies are plagued with problems of defining requirements, managing changing and evolving requirements, managing human and material resources, and maintaining adequate communication among individuals and organizational units [participants in the problem solving exercise]; and
- the net result is usually: cost overruns, delayed deliveries, systems that do not satisfy the users, and systems that are inadequately documented.

REFERENCES

1. Nadler, Gerald, "Systems Methodology and Design", Journal of Mechanical Engineering, September 1986, PP. 84-87.
2. Dos Santos, Brian L., "A Management Approach to Systems Development Projects", Journal of Systems Management, (Association for Systems Management), March 1986, PP. 35-41.
3. Sage, Andrew P., "Systems Engineering", (Notes from a presentation made at Goddard), February 6, 1989.
4. Steppel, S. et al, Digital System Development Methodology, Version 2.0, Computer Sciences Corporation, 1985.
5. Mumford, E. and Henshall, D., "A Participative Approach to Computer Systems Design", Associated Business Press, 1978.
6. Lackman, Michael, "Controlling the Project Development Cycle", Journal of Systems Management, (Association for Systems Management), February 1987, PP. 7-29.
7. Systems Engineering Management Guide, Second Edition, (Fort Belvoir, Virginia: Defense Systems Management College), December 1986.

5. FRAMEWORK FOR EVALUATING METHODOLOGIES

This chapter is divided into four major sections: Section 5.1 presents some very basic approaches to developing systems; Section 5-2 addresses some of the evolving theories and approaches; Section 5-3 discusses the criteria developed in Phase One for evaluating methodologies; and Section 5-4 summarizes the relevant information in the chapter.

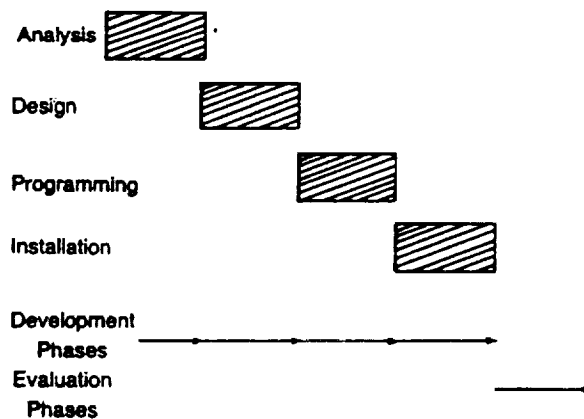
5.1 BASIC APPROACHES TO SYSTEMS DEVELOPMENT

This section describes the Traditional, Prototyping, and Incremental approaches to systems development and provides a very brief assessment of their effectiveness in handling different types of development projects.

5.1.1 The Traditional Approach

The traditional approach breaks up a project into distinct phases, such as: analysis, design, programming, and installation, to be undertaken in sequence.[1] Each phase must be completed before the next phase can begin. The work undertaken in each phase must be thorough, leaving no loose ends. The development process is thought of as an assembly line, allowing each phase to be undertaken by different personnel. Finally, no productive capability is installed until very near the end of the project. See Figure 5-1.

FIGURE 5-1: TRADITIONAL APPROACH TO SYSTEM DEVELOPMENT



[Reference 2]

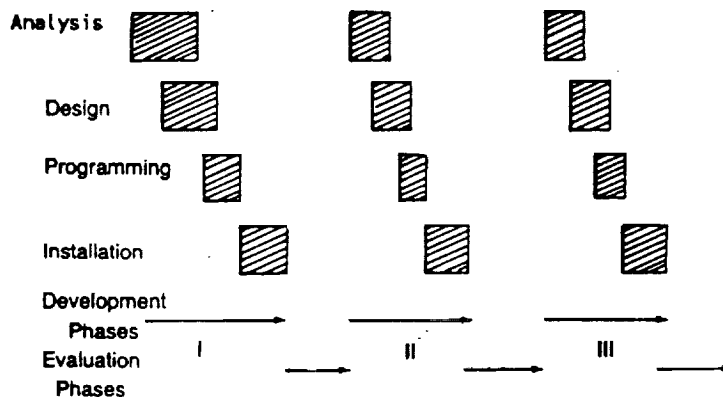
In this process it is essential to specify at the outset, precisely what will be done during the project. The approach works best when the problem is well-defined, the expected solution is highly structured, and is likely to remain so while the development is in progress. In practice this approach works well for a single type user who knows his needs, and whose needs are unlikely to change during the development. Good candidates for this approach are projects involving the replacement of an existing system that can be accomplished in six to twelve months.[2]

5.1.2 Prototyping Approach

The prototyping approach focuses on installing an inexpensive, experimental version (referred to as a prototype) of the system within a short period of time. The idea is that a prototype will be replaced by another version of the system, in succession, until an acceptable version is in place. See Figure 5-2. This iterative approach can involve several time phases, with each phase including some definition, design, and implementation activities, followed by an evaluation of the system's performance.

This approach is best suited for highly innovative systems. The key criteria for employing the prototyping approach are fuzziness in user requirements and a short project duration.[2]

FIGURE 5-2: PROTOTYPING APPROACH TO SYSTEM DEVELOPMENT



*This diagram illustrated three phases. They may be more than three phases, but there should be no less than two.

[Reference 2]

5.1.3 Incremental Approach

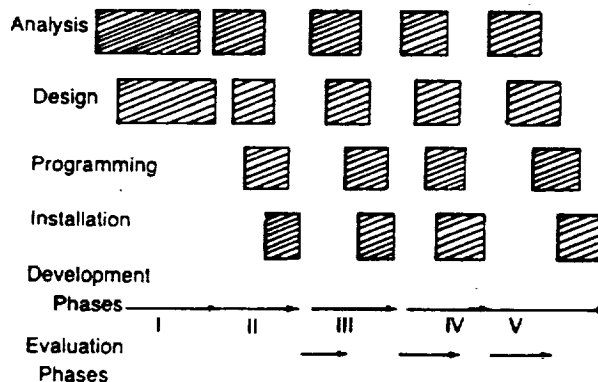
This approach focuses on designing an expandable system that can be installed in stages. At each stage, an additional user-operable capability is provided, i.e. a portion of the system that can be productively used is turned over to users. See Figure 5-3. In contrast, the traditional approach, produces paper products that have value to the developers, but have no value to the users. Like prototyping, an evaluation phase follows each development phase, other than the first phase, until the project is completed. The objective is to have the product of phase two operational within six months and new capabilities added at short intervals thereafter. Detailed design decisions are only made during the stage when the related capability will be installed.

This approach is particularly suited to developing large systems, of extended duration, that affect many diverse users. It is usually possible to plan an incremental project so that each phase after the first affects only one type of user. In this way, the complexity of the project is limited to dealing with a single user at a time.[2]

5.2 EVOLVING APPROACHES TO SYSTEM DEVELOPMENT

This section discusses work in progress by Warfield, Sage, and Fabrycky and Blanchard in the area of new theories, methods, and approaches to systems development.

FIGURE 5-3: INCREMENTAL APPROACH TO SYSTEM DEVELOPMENT



*This diagram illustrates five phases. There may be more or less depending upon the size and complexity of the system. However, there must be a minimum of three phases.

[Reference 2]

5.2.1 A Science of Generic Design

Warfield is proposing a Science of Generic Design as a response to the ever increasing need for larger and more complex systems with enormous and seemingly increasing potential for harm to humans and destruction of property.

He classifies systems as being of four classes. Class A consists of members that are clearly found in the physical sciences. Examples of which are radio, television, laser technology, semiconductor chips, and internal combustion engines. Class B consists of members that are sometimes referred to as "intellectual technology" or products of "artificial intelligence". Examples of which include computer software, textbooks about computer software, computer languages. Class C is comprised of a mix of members of Classes A and B, whose satisfactory performance depends on appropriate integration of these two classes into synergistic units. Examples of Class C include management information systems, management support systems, expert computer systems, hospitals, nuclear power plants, and banks. The fourth class of system is identified as "sociotechnical systems". These systems are comprised of technology and people, and depend on synergistic interaction of these two different kinds of entity for their satisfactory performance.[3, PP. ii-iii]

Warfield's Generic Design Theory (GDT) is based on drawing a distinction between two major concepts: generic and specific design. Generic design is derived from the observation that no matter what is being designed, certain kinds of creative and organizational efforts are necessary. Furthermore there is guidance from individual past experiences and other disciplines, such as engineering, anthropology, sociology, and psychology, which can inform and improve the activities of generic design. Specific design refers to the highly specialized knowledge and associated experience that individuals have developed and applied during particular design activities. Specific design knowledge and activities are not generally of interest to or readily understood by people outside of the specialized area, but the wise and skillful use of this knowledge and experience is indispensable to certain design activities. Working from these concepts, four postulates, three design laws, 13 design principles and several methodologies have been organized for the development of a science of design.[4, P. 42]

5.2.2 Life Cycle Engineering Design

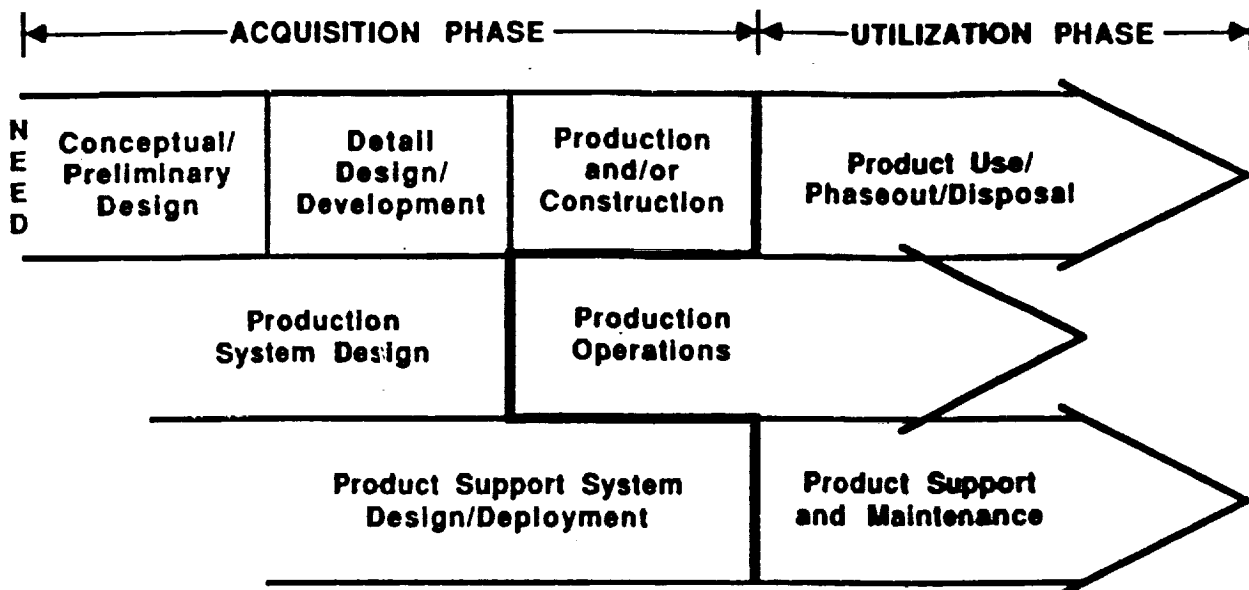
Fabrycky and Blanchard propose life cycle engineering design as an integration approach for bringing competitive products and systems into being in a way that minimizes their deficiencies and

life cycle cost.[5] This integration involves both design and development efforts to:

- Transform an operational need into a description of performance parameters and preferred product configuration through the use of an iterative process which includes functional analysis, synthesis, optimization, definition, design, test, and evaluation;
- Consider related technical parameters and ensure compatibility of physical, functional, and project management interfaces in a manner that optimizes the total product definition and design; and
- Integrate performance, producibility, reliability, maintainability, manability, supportability, and other "ilities" into the overall design process.[6]

A life cycle design approach for bringing competitive products and systems into being must go beyond consideration of the life cycle of the product itself. It must simultaneously embrace the life cycle of the manufacturing system as well as the life cycle of the product service system. Accordingly, there are three coordinated life cycles, progressing in parallel, as illustrated in Figure 5-4.

FIGURE 5-4: PRODUCT, PROCESS, AND SUPPORT LIFE CYCLES



[Reference 7]

The design approach which consists of six major phases is provided with additional details in Figure 5-5, and listed as follows:

- Conceptual Design,
- Preliminary Design (Advanced Development),
- Detailed Design and Development,
- Production and or Construction,
- Utilization and Support, and
- Phaseout/Disposal.[7]

5.2.3 Formulation, Analysis, and Interpretation Functions of Systems Engineering

Sage defines systems engineering in terms of its structure, function, and purpose as follows:

Structure

Systems engineering is management technology to assist clients through the formulation, analysis, and interpretation of the impacts of proposed policies, controls, or complete systems upon the need perspectives, institutional perspectives, and value perspectives of stake holders to issues under consideration.

Function

Systems engineering is an appropriate combination of the theories and tools, made possible through use of suitable methodology and systems management procedures, in a useful setting appropriate for the resolution of real-world problems, often of large scale and scope.

Purpose

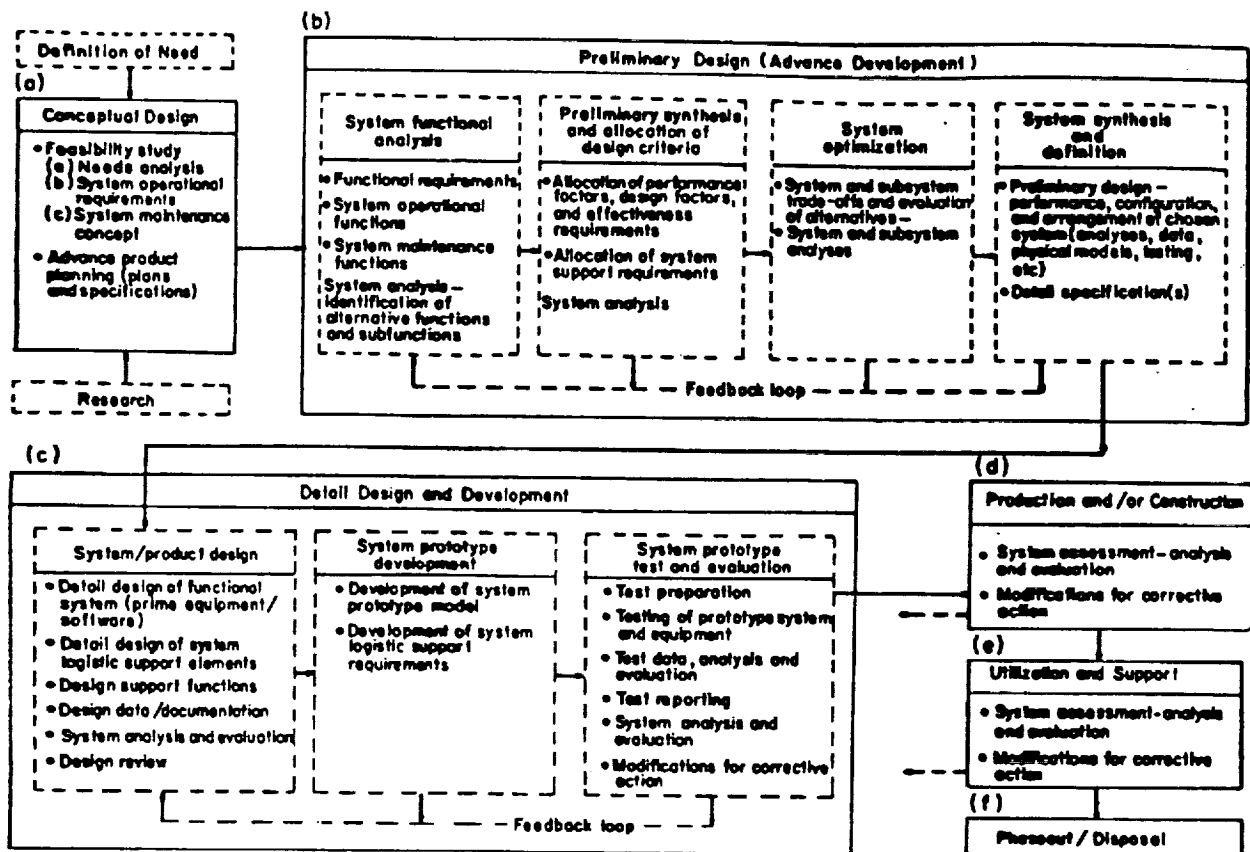
The purpose of systems engineering is information and knowledge organization that will assist clients who desire to develop policies for management, direction, control and regulation activities relative to forecasting planning, development, production and operation of total systems to maintain overall integrity and integration as related to performance and reliability.[8]

With these functions, he proposes a framework for systems engineering which consists of three fundamental levels (Figure 5-6), and Systems Management as an integral part of the systems engineering framework, as illustrated in Figure 5-7, below.

5.3 CRITERIA FOR EVALUATING METHODOLOGIES

One of the products of Phase One of this effort was a set of criteria for evaluating methodologies for potential application to the ND systems development projects.

FIGURE 5-5: LIFE CYCLE DESIGN ACTIVITIES AND INTERACTIONS



[Reference 7]

The set of criteria developed were divided into five major sub-sets:

- Structure of the Methodology,
- Flexibility,
- Accountability,
- Documentation of the Methodology, and
- Special Considerations of the User/Organization.

Because the word "systems" is used by people of different backgrounds to mean so many different things [the hierarchy of systems spans from the atom to the universe], the criteria in the category "Structure of the Methodology" were selected and used as a filtering mechanism to ensure the methodologies selected for more detailed evaluation were generally applicable to the type of projects. This desire to be in the "ball park" was balanced against the desire not to eliminate potentially applicable

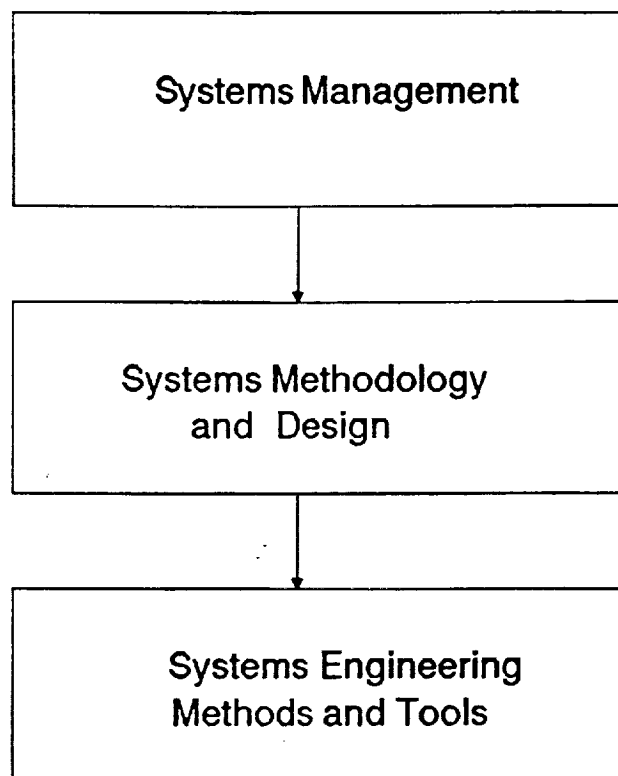
methodologies, prematurely, because the screening criteria are too rigorous.

Because the process of developing systems is very dynamic, flexibility is an essential feature of a "good" methodology. The criteria in the sub-set "Flexibility" were intended to test for flexibility in arenas, such as handling new information, managing resources, and adapting to different size and types within the same class of system development projects.

The criteria in the sub-set "Accountability" were selected to determine whether or not the methodology had incorporated mechanisms for managing its application, including documenting its activities and results, auditing, tracking, and maintaining the integrity of the process.

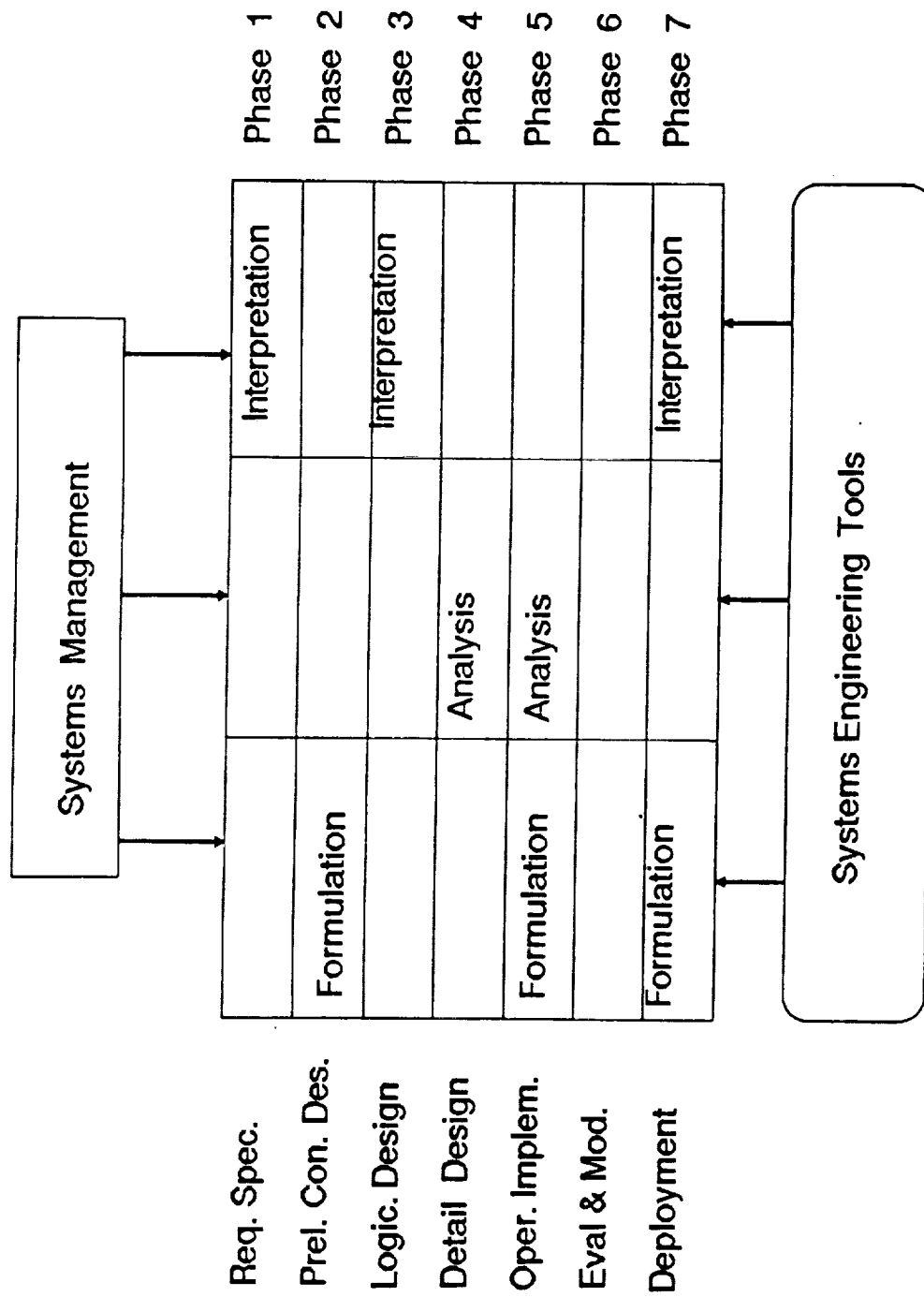
"Documentation" is concerned with the quality of the documents which describe the methodology including completeness, clarity, and level of detail provided.

FIGURE 5-6: THREE FUNDAMENTAL LEVELS OF SYSTEMS ENGINEERING



[Reference 8]

FIGURE 5-7: FRAMEWORK FOR SYSTEMS ENGINEERING



Again, because of the diverse use of the word "system", any meaningful evaluation of system methodologies must consider both generic and the specific requirements for the methodology. Thus, the final sub-set of criteria "Special Considerations of NASA" were intended to identify features in the methodologies that were of particular importance to the ND's projects. These methodological requirements were identified in Phase One of this effort. The evaluation criteria follow:

1. Scope and Structure of the Methodology

Does the methodology address activities that are likely to involve engineering work such as design, construction, installation, and operation?

Is it structured to handle large-scale or complex systems

Is it structured to handle at least one component that is extensively hardware?

Is the methodology partitioned into clearly defined and logical phases, processes, activities, or tasks that can be used as a basis for resource allocation and events such as the start or completion of phases that can be used as milestones or decision points?

2. Flexibility

Does the methodology accommodate systems of varying sizes and levels of complexity?

Does the methodology address ways of handling new information, feedback, or unforeseen circumstances (Iterative)?

Does the methodology allow for acquisition through a variety of approaches (procurement, development, etc.)?

Does the methodology allow maximum flexibility with time-allocation (scheduling) of resources?

Does the methodology address ways of identifying and selecting the best human and material resources to assign or allocate to its various phases of development?

3. Accountability

Does the methodology specify the documentation that is appropriate at different points during its application?

Does the methodology provide strategies and tools for communication and information exchange to ensure that all participants are aware of significant project decisions and have the most up-to-date information on the project status and activities?

Does the methodology specify management procedures to ensure that it has been applied as intended?

Does the methodology identify its intended users, class of systems, and scope of its intended applications?

Does the methodology address ways of addressing critical considerations such as national security, risk (environmental, evolving technologies), human safety, etc.?

4. Documentation of the Methodology

Is the methodology written clearly, precisely, completely, and at a level of detail that is appropriate for its intended users? Is it a good road map?

5. Special Considerations of NASA

Is the methodology fairly independent of organizational structure?

Does the methodology provide for the retention of key personnel throughout the life-cycle?

Does the methodology provide for the incorporation of requirements identified during the system analysis, design, or subsequent phases?

Does the methodology provide tools and techniques for predicting or projecting future requirements, through the planning horizon?

Does the methodology suggest strategies and techniques for designing and developing systems in the absence of specific requirements?

Does the methodology provide tools and techniques (including graphics and prototyping) for communicating among individuals and various organizations or organizational units working on major systems projects?

Does the methodology provide tools and techniques for redesigning and making major modifications to extend the useful life of a system in operation?

Keys identifies "plurality" and "uncertainty" as the two major factors which affect the choice of methodology. Plurality is concerned with the nature of the subjects: individuals, small groups, or whole organizations. The degree of plurality increases as the number of subjects increases. Uncertainty is concerned with the extent and quality of information available regarding the problem-situation. His methodology for methodology choice involves five phases designed to determine the levels of plurality and uncertainty/complexity, and to select methodologies that are appropriate for different combinations of plurality and complexity.[9]

Keys' effort focuses on the phases or tasks involved in selecting a methodology. His criteria for selecting a methodology include, primarily, considerations of plurality and complexity.

5.4 SUMMARY

The traditional approach is suitable for some of the sustaining engineering projects of the Networks Division, because the requirements are usually well defined, the projects are relatively small and limited in scope, and the duration is usually one year or less. The vast majority of ND's projects will require a skillful combination of all three design approaches. This has been recognized by the Mission Operations and Data Systems Directorate, and its current methodology, described in its Systems Management Policy, essentially follows the Incremental Approach, however, it incorporates baselining [derived from the Traditional Approach], and allows for prototyping of critical components.

Warfield's theory of Generic Design addresses some of the fundamental problems of large and complex systems design and presents some conceptual and methodological approaches to resolving these problems. Some of Warfield's methodologies, with possible application in the ND, are discussed in Chapter 14--Tools and Techniques.

Fabrycky and Blanchard's Life Cycle Engineering Design addresses two primary issues. The first seems more applicable to the development of products or systems that are produced in an assembly line type production process (e.g., automobiles). In these design problems an integral part of the process of designing the item is designing the plant that will produce the required quantity of such items and the facilities that will maintain those items over the life cycle. The second issue is that of developing approaches and techniques to optimizing performance over the entire life cycle of the item. Their concern is that current approaches involves using one set of optimization techniques for the design/development phase and using another set of techniques for the operations/maintenance phase. This piece-meal optimization is

quite likely to yield systems that are not globally optimized. The latter issue has very significant implications for the ND which, like most government organizations, seems to optimize on a much finer level--contract by contract optimization, and which seems to weigh cost above all else in the optimization process.

Sage defines systems engineering as management technology, and considers systems management to be an integral and very important part of the systems engineering process.

REFERENCES

2. Dos Santos, Brian L., "A Management Approach to Systems Development Projects", Journal of Systems Management, (Association for Systems Management), March 1986, PP. 35-41.
3. Warfield, John N., A Science of Generic Design, (In Publication).
4. Christakis, Alexander N., Keever, David B., and Warfield, John N., Development of Generalized Design Theory and Methodology, (Fairfax, Virginia: Center for Interactive Management, George Mason University), 1987.
5. Fabrycky, Wolter J., "Design for Life Cycle", Mechanical Engineering, Volume 109, No. 1, January 1987.
6. Thuesen, G. J., and Fabrycky, W. J., Engineering Economy, Seventh Edition, (Englewood Cliffs: Prentice-Hall, Inc.), 1989.
7. Fabrycky, W. J., and Blanchard, B. S., "Life Cycle Engineering Design", (Prepared for: the Handbook of Design Management).
8. Sage, Andrew P., "Systems Engineering", (Notes from a presentation made at Goddard), February 6, 1989.
9. Keys, Paul, "A Methodology for Methodology Choice", Journal of Systems Research, Vol. 5, No. 1, (Pergamon Press), 1988, PP. 65-76.
10. Fabrycky, W. J., "Systems Engineering: Life Cycle and Concurrent Design Methodology" (Notes from a presentation made to Goddard), March 6, 1989.
11. Blanchard, B. S., Engineering Organization and Management, (New Jersey: Prentice-Hall, Inc.), 1976.

12. Davidson, Frank P., Macro-Engineering and the Future: Management Perspective, Westview Press, 1982.
13. Banathy, Bela H., "Matching Design Methodology to System Type", Journal of Systems Research, Vol. 5, No. 1, (Pergamon Press), 1988, PP. 27-34.
14. Jackson, M. and Keys, P. "Toward a System of Systems Methodologies", Journal of Operations Research Society, Vol. 35, 1984, PP. 473-486.
15. Klir, George J., Architecture of Systems Problem Solving, (New York: Plenum Press)

6. REVIEW OF THE SYSTEMS MANAGEMENT POLICY OF MISSION OPERATIONS AND DATA SYSTEMS DIRECTORATE

The Systems Management Policy (SMP) is the primary methodology being used by the Mission Operations and Data Systems Directorate (MO&DSD) and its subordinate the Networks Division (ND). A summary of this policy follows.

It is the policy of the MO&DSD to apply consistent systems engineering and management practices to the development of all of its systems.

At MO&DSD, a system can be the end-product delivered at the completion of a project, and as such the development and application of systems engineering methodologies is within the broader contexts of "project or systems management."

MO&DSD defines a project as any flight project-supported system activity or institutional system activity with a definitive start and end date where a product is delivered upon completion, and categorizes projects into three levels as follows: Level I projects are typically inter-divisional and require directorate-level approvals. Inter-agency, inter-center, and inter-directorate projects for which MO&DSD has been assigned primary responsibility are designated Level I projects. Level II projects are typically intra-divisional, between branches within a division, and require division-level approvals. Level III projects are those conducted within a branch, and require branch-level approvals.

The SMP assigns the responsibility for planning, organizing, monitoring, and controlling the project to the project manager. Project managers are employees of MO&DSD who are trained and experienced in the field of project management.

Two basic structures have been established to assist in the systems development processes: (1) the Work Breakdown Structure (WBS) and (2) the system life cycle model.

6.1 THE WORK BREAKDOWN STRUCTURE

A WBS is a graphic (tree-structured) tool used to divide the total work of a project into logical and manageable tasks, sub-products, and/or phases. WBS depicts the hardware, software, data, and other related services that must be provided to develop a system. While the WBS may be structured to meet the requirements of the particular project, MO&DSD has established a typical WBS that would ensure that certain important considerations are not omitted during the project planning stage. The typical WBS

contains the following products and services, depicted in Figure 6-1.

- Project Management,
- Systems Engineering,
- Hardware,
- Software,
- System Test and Evaluation,
- Data,
- Training, and
- Operations and Maintenance (O&M).

6.1.1 Project Management

The project management element of the WBS refers to the administration and business aspects of the planning, organizing, directing and coordinating, controlling, and reporting activities related to accomplishing the project objectives.

The project planning process involves the definition of the work elements necessary to develop the required system; the relationship of these elements to the WBS; and the establishment of budgets and schedules for each defined work element.

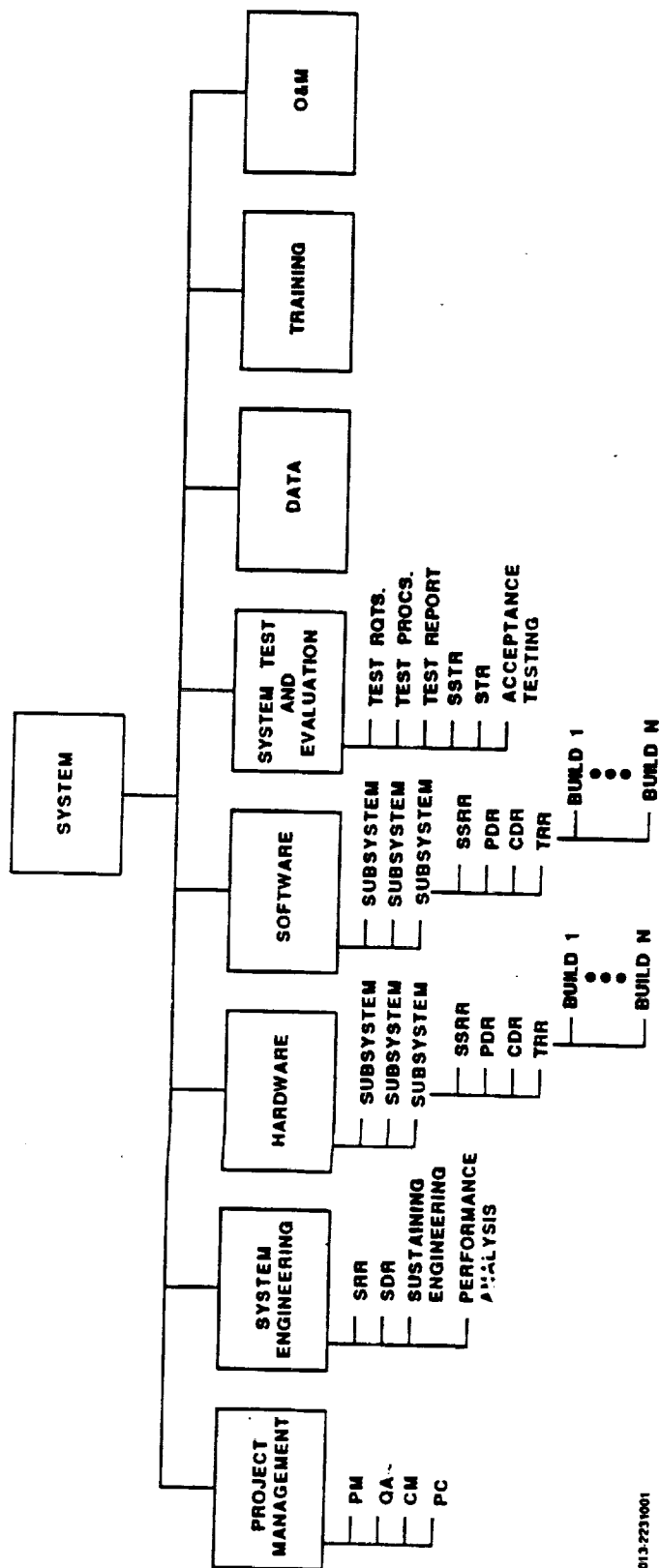
The project organization process involves the establishment of an organizational structure for implementing the project; assigning work elements, budgets, and schedules to each organizational unit; and formally documenting the overall plan and organization for MO&DSD approval.

Directing and coordinating are functions within the Project Management element of the WBS under control of the project manager which are vital to accomplishing project objectives. The coordinating function is particularly important in rather complex projects entailing several subsystems.

The project monitoring and controlling process involves the reporting and analyzing of cost and schedule status; the monitoring of the product performance against established budgets; and the quality assurance (QA) and configuration management (CM) of the project.

The Reporting process is one of the keys to the ongoing and successful implementation of the SMP. Suspected system problems are normally reported by operation personnel using problem report forms. The problem reports are logged, distributed, and assigned to a specific individual or manager for resolution in accordance with configuration control procedures. Performance reporting facilitates the evaluation of current progress as well as the prediction of future performance.

**FIGURE 6-1: WORK BREAKDOWN STRUCTURE
FOR A TYPICAL MOEDSD PROJECT**



0013-2231001

[Reference 1]

Also included in this WBS element are the project manager's and administrative function, the quality assurance function, the configuration management function, and the project control function.

A Project Management Plan is required at the completion of the planning process. The Plan details the management and technical approaches to the project, and includes considerations such as:

- Configuration Management,
- Quality Assurance,
- Management Reviews,
- Security, and
- Hardware and Software Management.

Project Management Plans require approval by the appropriate level of management within MO&DSD or Headquarters.

6.1.1.1 Configuration Management

Configuration management is generally applied over the development, integration and tests, and operation phases of the life cycle. The scope and formality of the CM will depend on project type, system size, system complexity, and the risks associated with system development. The CM of a system includes control of the system, configuration identification, status accounting, Configuration Control Board (CCB) audits and traceability.

(1) Configuration Management Plan

The purpose of this plan is to apply CM throughout the system development, integration and test, and operation phases of the life cycle in order to achieve the following objectives:

- Establish system baseline,
- Maximize control at the responsible level,
- Maximize responsiveness and minimize formality,
- Identify items that will be subjected to configuration control,
- Provide management flexibility,
- Provide traceability,
- Ensure thorough coordination of proposed design change to the established baseline,
- Provide uniform reporting and documentation, and
- Ensure management visibility of technical changes.

(2) Configuration Audits

The purpose of a configuration audit (CA) is to ensure compliance with approved CM plans, procedures, and configuration documentation. CA's are conducted on CCB-controlled hardware, software, documentation, and procedures. Two types of CA's are performed. The first is audits of baseline and configuration items, such as:

- Requirements baseline,
- Functional baseline,
- Allocated baseline,
- Development baseline,
- Design baseline,
- Product baseline,
- Operational baseline,
- Configuration management code, and
- Security baseline.

The second is audits of CCB procedures and methods, such as:

- Forms control,
- Status accounting, and
- Documentation library control.

A schedule of CA's is prepared as part of the CM plan; however, unscheduled audits can be conducted on complex projects, involving high technological risks or scheduling uncertainties. Formal reports are prepared to document the results of each CA.

6.1.1.2 Quality Assurance

Quality Assurance activities ensure that standards and practices are established and put into effect; that inspections and audits are done; and that all assurance activities are carried out according to needs and schedule.

6.1.1.3 Management Reviews

Management reviews are done at the completion of each phase. Here all verification is done with the aim of keeping the project on track with regard to cost, time and functionability of the project. At this point decisions are made concerning any major changes that may be required for the completion of the project.

6.1.1.4 Security

Security considerations within MO&DSD's PMP are usually of paramount importance. This of course is due to the fact that NASA

operates on the leading edge of technology, with the US desire to stay ahead of its competition, most information is released on a "need to know" basis. Security considerations are governed by the Designated Approval Authority (DAA), a designated Department of Defense or GSFC official responsible for ensuring that specified systems/facilities which store and process classified information, meet and maintain their prescribed security requirements.

6.1.1.5 Hardware and Software Management

Hardware/Software management governs all equipment that is purchased, leased or built and delivered to fulfill the requirements of the project. The management system involved is thus responsible for the procurement of all hardware/software necessary for successful completion of the project.

A contractor's work performance on a MO&DSD's project is measured by cost, schedule, and technical factors. In measuring performance dollar values are assigned to scheduled milestones within the WBS. These cumulative values are identified as the budgeted cost of work scheduled. The budgeted cost of work performed (BCWP) is the cumulative budgeted value of milestones achieved. A comparison of these measures gives an indication of how well the project is adhering to the schedule. The actual cost of work performed represents the actual cumulative cost expended to achieve the milestone. This measure, compared with the BCWP, produces a indication of cost-effectiveness to that milestone in the WBS. The technical performance is determined during program reviews of the project.

6.1.2 System Engineering

The system engineering WBS element refers to the management and technical efforts related to directing and controlling a totally integrated engineering effort for the system. System engineering includes system requirements analysis, analysis and system design, the sustaining engineering to support ongoing performance analyses and design trade-offs, and the definition of system interfaces. Various trade-off analyses used to arrive at the optimum system design, considering the project constraints of cost and schedule, are included. Systems engineering also includes the efforts related to controlling requirements and maintaining the traceability of all requirements throughout the development part of the system life cycle.

6.1.3 Hardware

The hardware WBS element covers to all equipment that is purchased, leased, or built and delivered to fulfill the

requirements of the project. All equipment should be broken down at the second level of the WBS by hardware subsystem. It also includes all maintenance cost associated with this equipment prior to delivery. Each hardware subsystem should be further broken down into:

- Subsystem requirements analysis,
- Subsystem architecture design,
- Subsystem detailed design,
- Subsystem logistics system analysis,
- Subsystem implementation, and
- Subsystem integration and test.

6.1.4 Software

The software WBS element refers to all software that is purchased, leased or developed, and delivered to fulfill the requirements of the project. All software should be broken down at the second level by software subsystem. If the software is to be developed in multiple builds or releases, this should be represented at the next lower level of the WBS. This WBS element also includes the responsibilities of the data base administrator. Each software subsystem should be further broken down into:

- Subsystem requirements analysis,
- Subsystem architecture design,
- Subsystem detailed design,
- Subsystem implementation, and
- Subsystem integration and test.

6.1.5 System Test and Evaluation

The system test and evaluation WBS element is concerned with the efforts related to independent system testing and formal system acceptance testing. This WBS includes the development of test requirements, test procedures, and test reports.

6.1.6 Data

The data WBS element refers to the effort required to acquire, type, edit, proof, illustrate, reproduce, pack, and ship all documentation required by the project. Specifically excluded from this WBS is the effort to write the various documents. These efforts are included in the WBS elements where technical work is accomplished.

6.1.7 Training

The training WBS element includes the training services, materials, and devices used to facilitate instruction through which the O&M personnel will acquire sufficient knowledge and skill to operate and maintain the system. It includes all efforts associated with the design, development, and execution of the training courses.

6.1.8 Operations and Maintenance

The Operations and Maintenance WBS element refers to efforts involved in supporting the system, following system acceptance and turnover.

6.2 THE SYSTEM LIFE CYCLE MODEL

The system development life cycle model developed by MO&DSD consists of eleven phases, as depicted in Figure 6-2, and listed as follows:

- Concept and Project Definition,
- System Requirements Analysis,
- System Analysis and Design,
- Subsystem Requirements Analysis,
- Subsystem Architecture Design,
- Subsystem Detailed Design,
- Subsystem Implementation,
- Subsystem Integration and Test,
- System Integration Test,
- System Acceptance Test, and
- Operation and Maintenance.

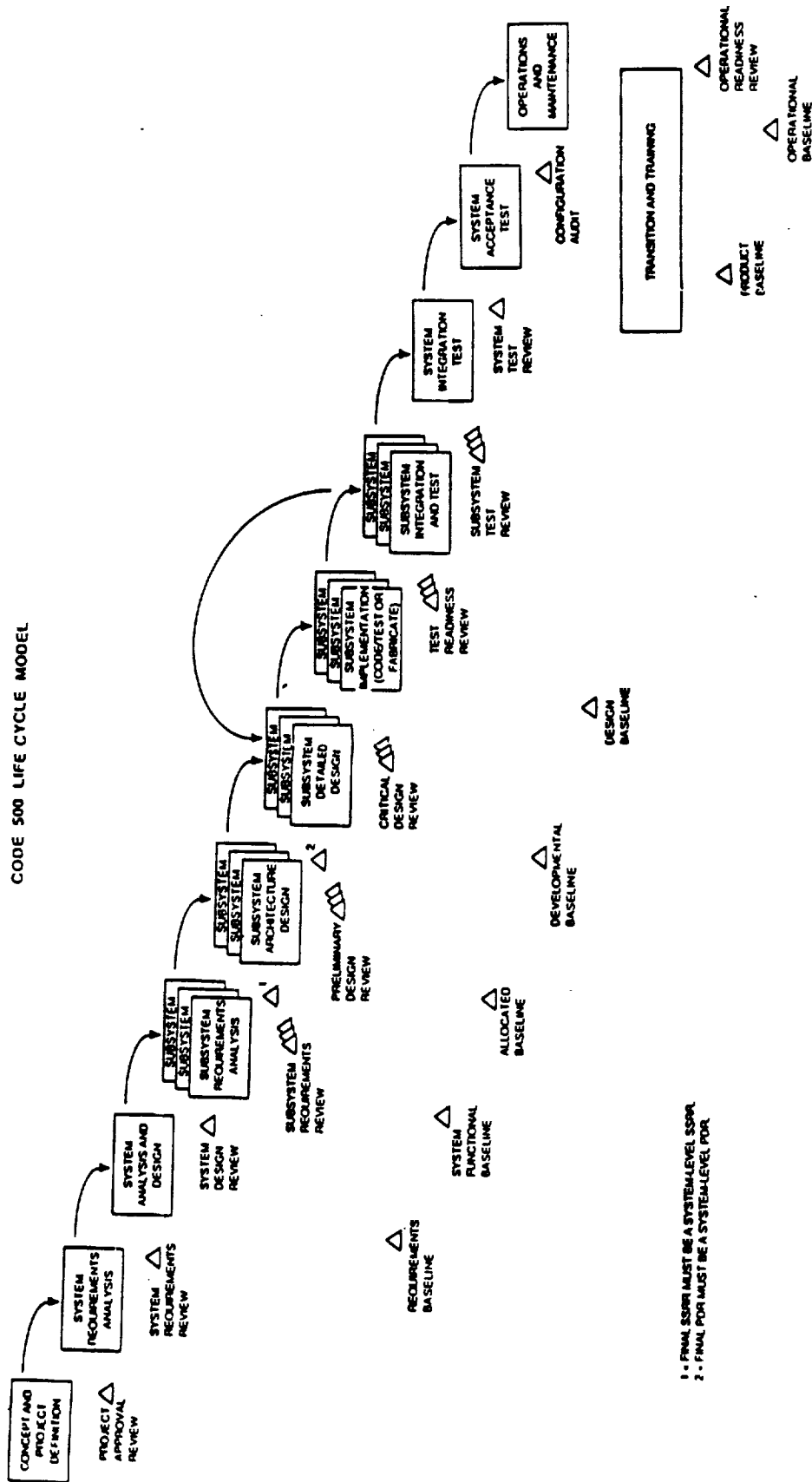
MO&DSD allows some flexibility in specifying the phases of the life cycle model for a particular project; such as: prototyping of critical subsystems, iterating through certain phases instead of considering them as strictly sequential, and the elimination of non-essential phases in developing software systems.

The documentation requirements for each phase of the life cycle is presented in Figure 6-3.

6.2.1 The Concept and Project Definition Phase

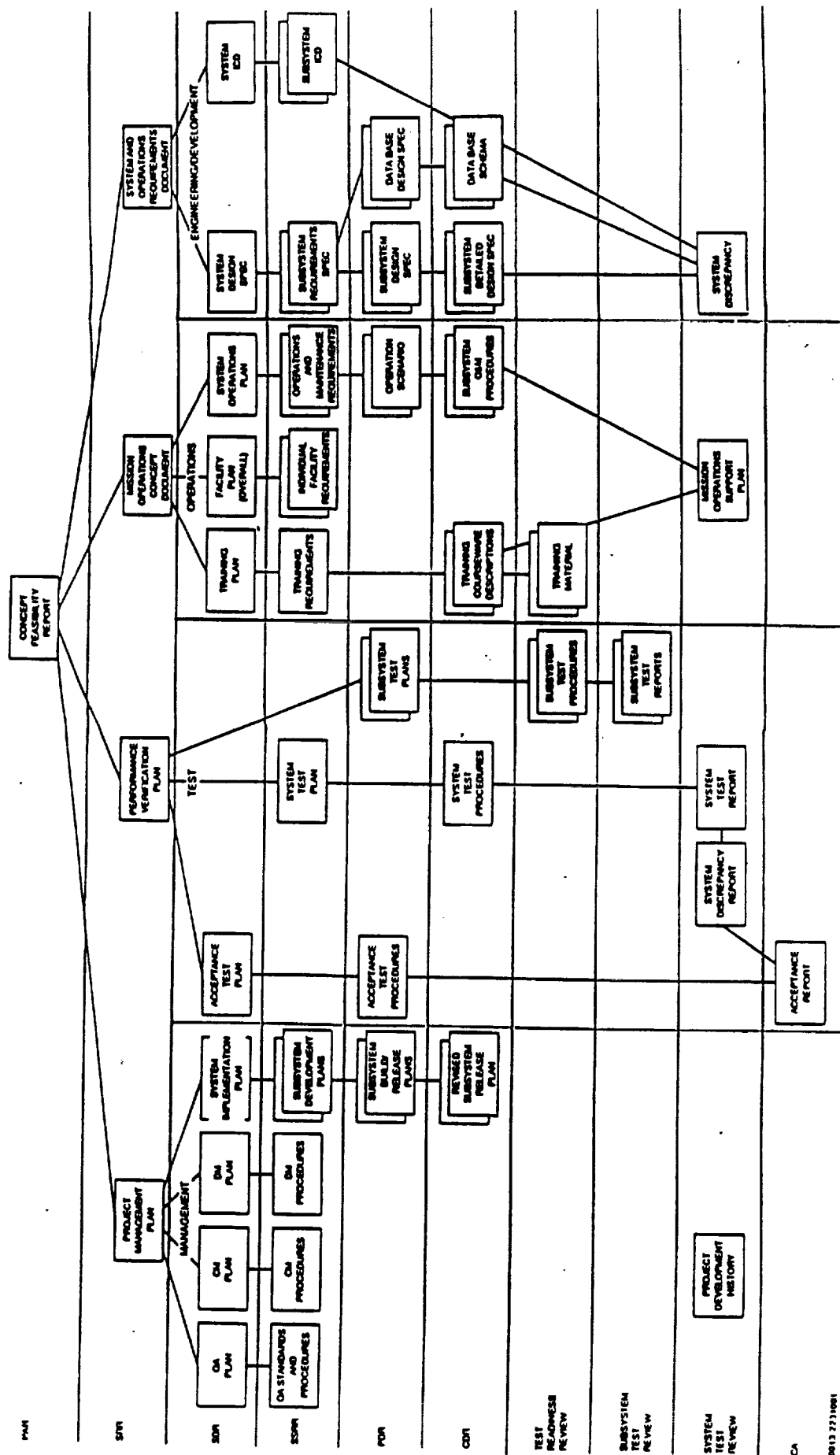
This phase starts with a support instrumentation requirements document, a memorandum of understanding, the mission objectives and the mission specifications and constraints.

FIGURE 6-2: ELEVEN PHASES OF THE SYSTEMS DEVELOPMENT LIFE CYCLE



[Reference 1]

FIGURE 6-3: DOCUMENTATION REQUIREMENTS FOR LIFE CYCLE



[Reference 1]

A feasible concept (both technical and economic) of problem solution must be generated for use as a basis for system requirements analysis. The concept should be based on an initial analysis of requirements and some preliminary trade studies. This step provides an initial cost/resource estimate using a conceptual scope. The preliminary concept must be documented in sufficient detail, along with key issues and trade study results, so that critical technological limitations and key system cost drivers are specified. If major procurement is envisioned, the procurement approach should be included.

This phase culminates in a project approval review that evaluates the "draft" concept feasibility report and, for large projects, the "preliminary" PMP and the "preliminary" systems and operations requirements document (SORD).

A review of the purpose of the system, background of the mission objectives, a summary of preliminary resources estimates, and a high level preliminary system requirements analysis should be presented. A life cycle cost analysis should be included along with an associated risk analysis.

6.2.2 The Systems Requirements Analysis Phase

This phase evaluates the mission objectives, derives the mission operations concepts, establishes the overall test strategy and develops the overall system requirements. This phase develops functional flows of the system, conducts system implementation trade studies, develops "preliminary" system interface control documents, and completes the PMP. This phase is completed when a system requirements review has been successfully completed and a "draft" PMP, SORD, test strategy, and mission operations concept document have been reviewed.

6.2.3 The System Analysis and Design Phase

This phase allocates the system requirements to subsystem. System performance analyses, make vs. buy studies, QA and CM plans, operational analyses, and acceptance test plans is completed during this phase. If a non-advocacy review is required on the project, it should occur during this phase. The system design review marks the completion of this phase.

6.2.4 The Subsystem Requirements Analysis Phase

This phase is conducted for each subsystem identified in the system design specification. Functional and performance analyses of the subsystem requirements are performed in this phase. The data base requirements must also be analyzed. For each subsystem,

requirements reviews are required to signal the completion of this phase.

6.2.5 The Subsystem Architecture Design Phase

This phase allocates all subsystem requirements to specific components of the design. For the hardware, equipment layout and preliminary drawings to the functional block diagram level are designed and documented. For software, design drawings showing functions allocated to tasks and modules down to the unit level are completed along with data base file designs. For subsystems with significant operator interactions, the preliminary display formats must be included in the subsystem design specifications.

To complete this phase, a separate preliminary design review should be conducted for each subsystem and must address the performance issues across all subsystems and its integrity with the overall system design.

For software, this review focuses on the evaluation of the progress, consistency, and technical adequacy of the selected design and test approach, and on the establishment of compatibility between software and preliminary design.

6.2.6 The Subsystem Detailed Design Phase

This phase performs the detailed design of all components (hardware and software) of the system. The "code to" or "fabricate to" drawings must be completed for each subsystem. This includes the design of the software and the allocation of functions to the unit level. This phase, along with the subsystem implementation and subsystem integration and test phases, are repeated for each build of the system. If these three phases require more than six months to complete, the project is broken into builds.

For software, this review focuses on the determination of the acceptability of the detailed design, performance, and test characteristics of the design solution.

6.2.7 The Subsystem Implementation Phase

For software, this phase consists of the coding and unit testing of all units to be developed in the build. For hardware, this phase covers the fabrication and unit testing of all components required in the build. Subsystem test procedures and training materials are required within this phase. The test readiness review, which completes this phase for each subsystem, involves a review of the documentation and the results of unit

testing to ensure that the software and hardware components are ready for integration testing.

6.2.8 The Subsystem Integration and Test Phase

This phase involves the integration of the individual software or hardware elements of the subsystem. This phase is accomplished once for each subsystem of each build. Subsystem test procedures are executed and the results documented in the subsystem test reports. This phase is completed when a subsystem test review is completed.

6.2.9 The System Integration Test Phase

This phase includes the integration of all subsystems on a build-by-build bases as subsystems are completed. System test procedures are executed and the results documented in a system test report. The mission operations support plan and the "as built" system description, including the final version of all detailed design documents, should be completed in this phase.

This phase is complete when a system test review has been conducted. Verification that all system tests were successfully executed should be accomplished. Complete documentation describing the system's operations and the "as built" hardware and software should be completed in this phase. For the final build of the system, the project development history and system discrepancy report describing all outstanding deviations and problems in the delivered system must be completed.

6.2.10 The System Acceptance Test Phase

This phase consists of the system testing by an independent team after the release of the system including the operating procedures and all software and hardware. The results of acceptance testing must be documented in an acceptance report.

This phase is complete when a CA has been performed to ensure that all required software, hardware, operational procedures, and documentation exist and are complete prior to being sent to the operations team.

6.2.11 The Operations and Maintenance Phase

This phase involves the full operations of the system and the normal maintenance of all subsystems.

6.3 SUMMARY

A detailed discussion of the methodological needs of the ND was presented in the Phase One report. In summary, however, the methodology provides the work breakdown structure, the life cycle phases and the reporting requirements. It does not provide a detailed set of steps or tasks necessary to successfully complete the project. The Project Manager is expected to develop such tasks.

The methodology relies very heavily on the experience and competence of the project manager in deciding what should and what should not be included in the project management plan, detailing the tasks to be performed, and the contents of specified reports. This should not be a major problem, when one considers that the methodology was developed in-house, with the developers being some of the primary users and with the developers having first-hand knowledge of the skills and capabilities of their counterpart users. The problem may intensify as some of the more experienced ND project managers retire, and they are replaced by less experienced staff.

The Project Manager is required to tailor the methodology to his particular project, and document the project methodology in the Project Management Plan. The methodology does not address acquisition through the procurement process, which is a major practice of the ND. The documentation of the methodology provides no discussion of special organizational structures, staffing or acquisition of other resources for the project. While the methodology identifies the types of documents needed at different points in the development process, with the exception of an outline of the Project Management Plan, it provides no information on the contents of these reports.

The Project Manager specifies the communication and reporting requirements in the PMP.

REFERENCE

1. Mission Operations and Data System Directorate, System Management Policy, (Greenbelt, Maryland: NASA Goddard Space Flight Center), 1986.

7. REVIEW OF HUGHES SYSTEMS ENGINEERING METHODOLOGY

The HUGHES Aerospace Program life cycle model consist of five distinct phases as shown in Figure 7-1. They are:

- Concept Definition;
- System Design;
- Detailed Design;
- System Integration and Test, and
- Production.

The systems engineering process, illustrated in Figure 7-2, is applied during these five phases. The steps and tasks associated with the systems engineering process are summarized as follows:

Step One: Defining Requirements

- Collect the Requirements,
- Specify the Requirements, and
- Select Verification Methods;

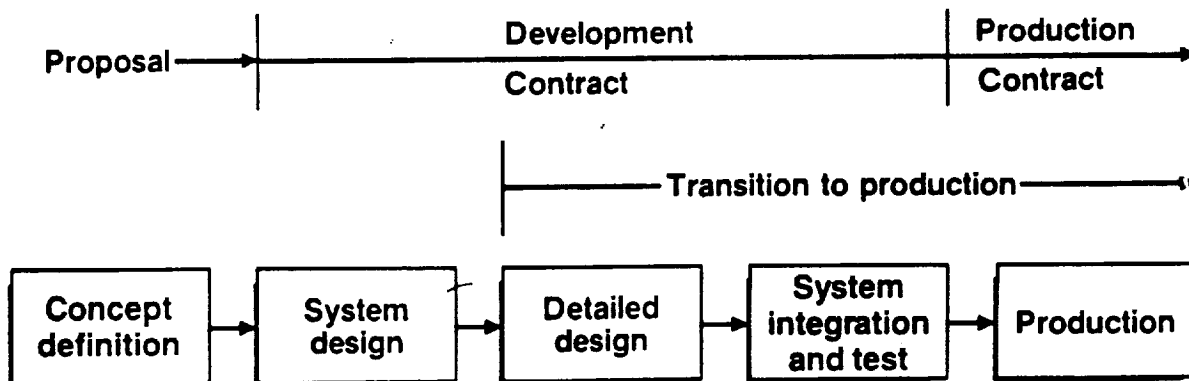
Step Two: Developing a Design

- Model and Simulate the Design,
- Define and Control Interfaces,
- Integrate Specialty Disciplines,
- Manage Resource Margins,
- Review the Design, and
- Control Design Changes; and

Step Three: Verifying Performance

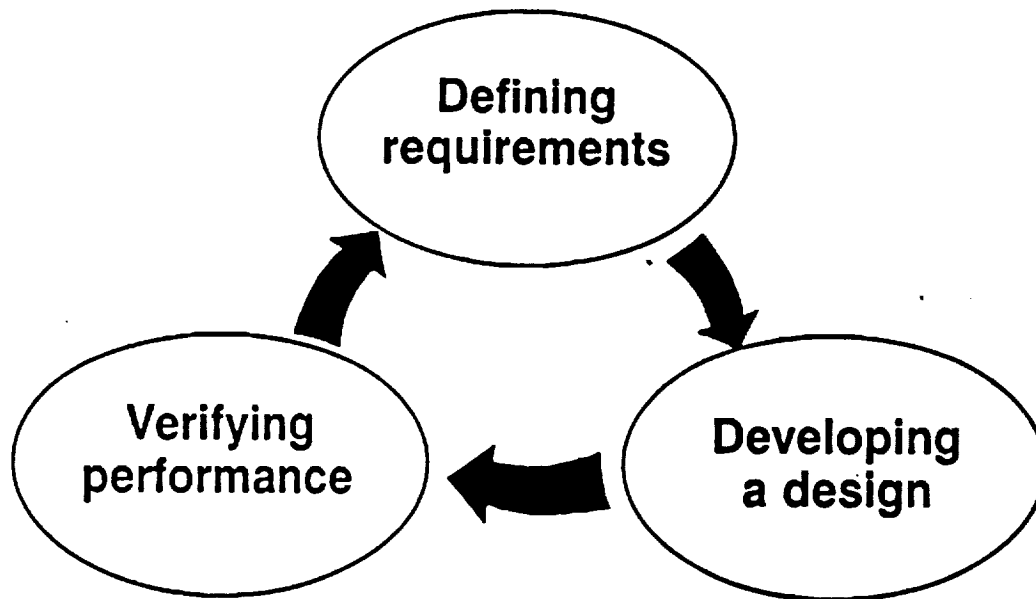
- Test the Design and
- Selling off the Product.

FIGURE 7-1: PHASES OF HUGHES SYSTEMS DEVELOPMENT PROCESS



[Reference 1]

FIGURE 7-2: SYSTEMS ENGINEERING AS A CONTINUAL PROCESS



[Reference 1]

7.1 CONCEPT DEFINITION

The customer requirement and technology base form the inputs to this phase. The customer requirement, stated in the form of a specification or operation scenario, is then acted upon by system engineers, developing them into sufficient detail to ensure a complete understanding of customer needs. The system concept or baseline is then developed with the available technology. The baseline is a system that is producible at reasonable risk, with the level of risk being a function of the maturity of the technology.

The result of this phase includes: documentation of the defined requirements in the form of system specifications and external interface requirements; description of how the activities are going to be managed (documented in a systems engineering management plan); a design baseline, documented with a description of the system and a concept of operation; and a proposal--a document describing the system to be provided, supporting analysis to validate the expected performance, management approach and price.

7.1.1 Concept Definition Phase Inputs

- Customer Requirements and
- Technology.

7.1.2 Process Characterization

- Requirements Definition:
 - Gathering,
 - Refining, and
 - Analyzing;
- Design Synthesis:
 - Selection of Elements,
 - Arrangement of Elements,
 - Control of Elements, and
 - Allocation of Requirements to Elements;
- Design Evaluation:
 - Will Elements/Arrangement meet requirements?

7.1.3 Concept Definition Phase Outputs

- External Interface Requirements,
- System Requirements Specification,
- Master Test Plan,
- Systems Engineering Management,
- Proposal, and
- Design Baseline (System Description and Concept of Operation).

7.2 SYSTEM DESIGN

This phase is initiated by the issuance of a contract by the customer for the development of a system or system segment. The phase concludes with requirements being given to the subsystem design area.

The primary activities during the system design phase include those necessary to begin the program. The system design submitted in the proposal will be updated to reflect the requirements contained in the negotiated contract. Request for Proposals (RFPs) are written for items that are to be subcontracted and the proposals submitted in response are evaluated. Preliminary design reviews are held at the system and subsystem levels. Design responsibility is delegated to internal organizations by work assignment delegations and to subcontractors by subcontracts.

7.2.1 System Design Phase Inputs

- Proposal:
 - Design Baseline,
 - Management,
 - Program Plans, and
 - Test Plans; and
- Contract:
 - Terms and Conditions,
 - Statement of Work,
 - Contract Data Requirements List,
 - System Requirements Specification,
 - External Interface Requirements, and
 - Deliverables List.

7.2.2 System Design Phase Activities

- Update Proposal Baseline,
- Systems Analysis,
- Requirements Allocation,
- Subcontract RFPs and Proposals,
- Environmental Analysis, and
- Preliminary Design Reviews.

7.2.3 System Design Phase Output

- System Segment Specification,
- Subsystem/Unit Design Requirements,
- Work Breakdown Structure,
- Schedules,
- Subcontracts,
- Environmental Requirements, and
- Work Assignment Delegations.

7.3 DETAILED DESIGN

The detailed design can be divided into three sub-phases. The first phase is project startup. It begins after systems engineering has partitioned the system into units, and ends with a conceptual design review. The second phase is the design phase, during which the actual design of the hardware occurs. This phase is characterized by many design reviews. The third phase includes the first article build and test. Upon completion of this phase, the hardware is delivered to the next level of integration.

Detailed design of a subsystem is characterized by the top down allocation of requirements to successively lower levels, the

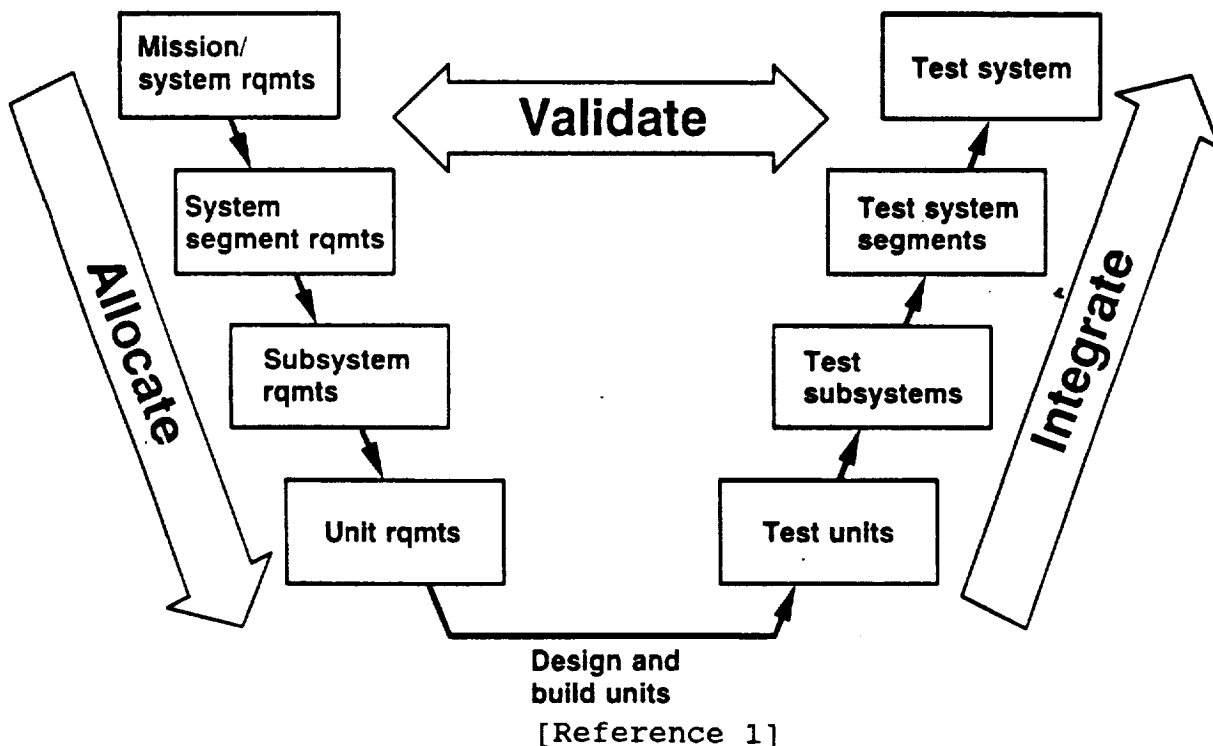
bottom up design/development process, and attention to details and a temporary task oriented organization.

Systems engineering after thorough analysis, allocates design requirements for each unit and provides them to the design engineering organization. Design engineering further allocates requirements into sub-units or modules and assigns them to engineers or engineering teams. This process is continued down to the component level. The design process then starts from the bottom up to develop first components, then units, and finally, subsystems to meet the specified requirements. See Figures 7-3 and 7-4.

7.4 SYSTEM INTEGRATION AND TEST

This phase of the development contract is actually a distributed task that begins during system design. The systems engineering approach ensures that when integration of the system finally begins, the problems encountered are minimized and a smooth transition to production can continue. The integration and test phase continues to support the production contract and plays a major role in all sustaining and follow on efforts. The major tool used by HUGHES in this phase is the "Integration and Test Plan". This plan is divided into three areas as shown below:

FIGURE 7-3: THE DESIGN PROCESS: ALLOCATE, INTEGRATE, VALIDATE

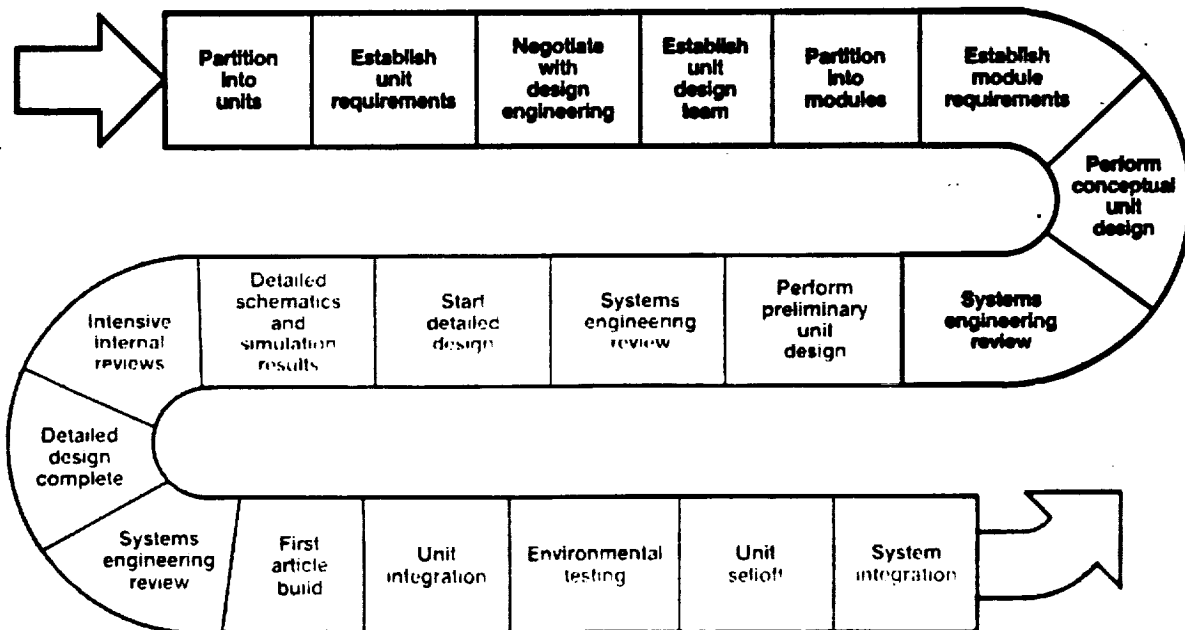


- Establishing Test Results
 - Customer,
 - System Design, and
 - Detailed Design.
- Test Preparation
 - Procedures,
 - Special Test Equipment, and
 - Facilities.
- Test Readiness Reviews

7.5 PRODUCTION

The release of the production drawings is a significant milestone in production startup. It signals the beginning of all the procurement activities that support the production build. When the drawings are released, the configuration management system ensures that at all times the hardware/software configuration is known and controlled. Stress screening plans, failure reporting, and corrective action systems are implemented in production. The final product sell off is conducted with a customer approved procedure prior to shipment. The inputs, activities, and output of the Detailed Design, the System Integration and Test, and the Production Phases are combined below:

FIGURE 7-4: THE DETAIL DESIGN PATH



[Reference 1]

7.5.1 Inputs to the Production Phase

- Detailed Subsystem/Unit Design Requirements,
- Company Standards and Practices,
- Program Plans and Schedules,
- Cost Goals, and
- Environmental Requirements.

7.5.2 Activities of the Production Phase

- Detailed Hardware/Software Design,
- Developmental Hardware Fabrication,
- Subsystem/Unit Testing,
- System Integration and Test,
- Production Startup,
- Configuration Management,
- Environmental Stress Screening,
- Reliability Monitoring,
- System Acceptance Testing, and
- Production Sustaining Activities.

7.5.3 Outputs of the Production Phase

- Tested/Certified System,
- Production Data Package, and
- System Test Procedures.

7.6 SUMMARY

The systems engineering process, of Defining Requirements, Developing a Design, and Verifying Performance, conducted at several points in the development cycle assures that the delivered system will satisfy the requirements. A Change Review Board ensures that only essential changes are made to the baseline and that the necessary information is disseminated. The change review process can also accommodate changes to the requirement that are within the scope of the contract with the customer. At HUGHES, the systems engineering function cuts across all phases of the development cycle, and a process has been developed to ease the transition from design to production. The methodology addresses procurement from the standpoint of a prime contractor offering systems engineering and systems development services to a customer and from the standpoint of a customer subcontracting services to a sub-contractor.

REFERENCE

1. Systems Engineering Seminars for General Motors, HUGHES Aircraft Company, 1987.

8. REVIEW OF COMPUTER SCIENCES CORPORATION DIGITAL SYSTEMS DEVELOPMENT METHODOLOGY

The Computer Sciences Corporation (CSC) Digital System Development Methodology (DSDM) was developed in response to the following problems:

- Difficulty in measuring the true status of the development effort accurately, especially when software is a principal element of the system.
- Implementation of design whose poor quality does not surface until the finished product is either subjected to final testing or installed for operation.
- High life-cycle costs resulting from a system that was not designed for reusability or maintainability.

The methodology applies to systems consisting of both hardware and software, however, it assumes that the hardware components will be purchased, while the software components may be either purchased or developed in-house. DSDM consists of five developmental phases as follows:

- Requirements Definition,
- Design,
- Implementation,
- Integration and Test, and
- Turnover.

Figure 8-1 summarizes the plans, products, reviews, and baselines that correspond to these phases of the development process.

8.1 REQUIREMENTS DEFINITION

The purpose of the system requirements definition phase is to collect, analyze, and document the system requirements. Also, to develop and document the system acceptance criteria and development plans. See Figure 8-2.

8.1.1 Activities of the Requirements Definition Phase

- Identify and Collect Requirements:
 - Functional,
 - Performance,
 - Operational,
 - Facility,

**FIGURE 8-1: SUMMARY OF
PLANS, PRODUCTS, REVIEWS, AND BASELINES FOR DSDM**

PHASE	REQUIREMENTS DEFINITION	DESIGN	IMPLEMENTATION	INTEGRATION AND TEST	TURNOVER
PLANS	<i>System Development Plans</i>	<i>System Test Plans System Implementation Plans</i>	<i>Operations</i>	—	—
MAJOR PRODUCTS	<i>System Specification System Acceptance Criteria</i>	<i>System Design Specification (includes hardware, software, interface, and operational specifications) Performance Model</i>	<i>Procured Components Developed Components Facility Operations and Maintenance Documentation Training Materials</i>	<i>Test Specifications and Procedures Accepted System</i>	<i>Operational System</i>
REVIEWS	<i>System Requirements Review</i>	<i>System Design Review</i>	<i>Software Specification Review Preliminary Design Review Critical Design Review</i>	<i>Functional Configuration Audit</i>	<i>Physical Configuration Audit</i>
BASELINES	<i>Functional</i>	<i>System Design</i>	<i>Allocated and Development</i>	<i>Product</i>	<i>Operational</i>

[Reference 1]

-
- Communication, and
 - Security;
 - Verify and Analyze Requirements;
 - Specify Requirements;
 - Develop System Concept;
 - Plan System Development; and
 - Develop System Acceptance Criteria.

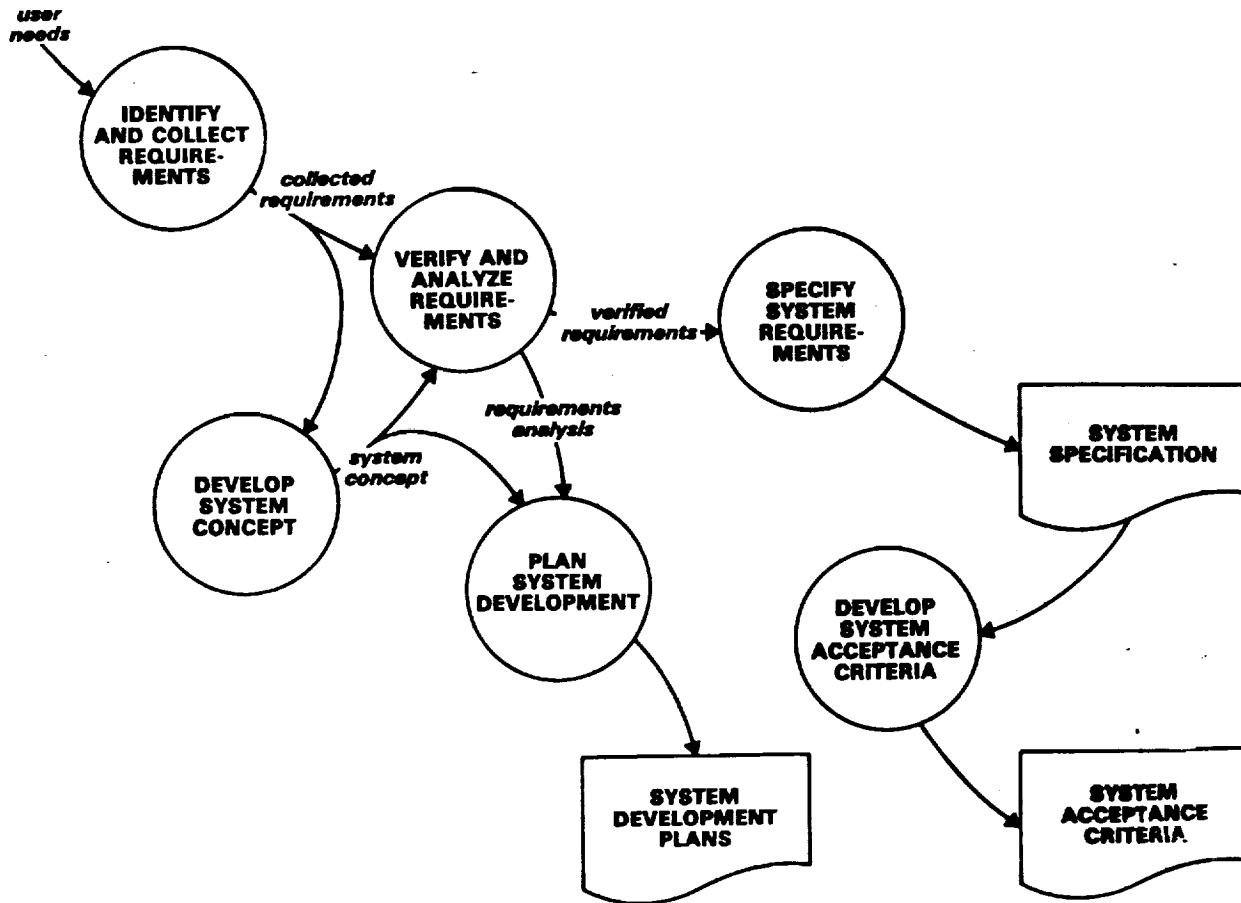
8.1.2 Products of the Requirements Definition Phase

- System Specification,
- System Acceptance Criteria, and
- System Development Plans:
 - Project Management Plan,
 - Systems Engineering Management Plan, and
 - Product Assurance Plan.

8.2 DESIGN

The primary goal of the system design phase is to identify the most cost-effective architecture and system configuration that will meet all requirements in the system specification (the functional Baseline), and will allow for future expansion. The primary output of this phase is the system design specifications which specify the

FIGURE 8-2: THE REQUIREMENTS DEFINITION PROCESS



[Reference 1]

system requirements and constraints to be met by each hardware, software, and facility configuration item (CI) and by the operational procedures.

8.2.1 Inputs to the Design Phase

- System Specification, and
- System Acceptance Criteria.

8.2.2 Activities of the Design Phase

- Develop and Analyze Alternative System Architectures,
- Develop System Configuration,
- Develop High-Level System Designs,
- Analyze and Select Best Design,

- Develop System Level Design Specifications,
- Establish and Run System Performance Model, and
- Plan System Implementation.

8.2.3 Outputs of the Design Phase

- System Test Plans;
- System Implementation Plans:
 - Procurement,
 - Material Management,
 - Facility,
 - Logistics, and
 - Training; and
- System Design Specifications:
 - Detailed Requirements Allocated to each Hardware and Software CI and to Manual Operations,
 - Performance and Resources Constraints on each CI,
 - Interface Requirements to be met by each CI, and
 - Operational Concept for the System Design.

8.3 IMPLEMENTATION

The primary purpose of the system implementation phase is to implement the system configuration defined in the system baseline. The implementation phase often begins before the design phase ends, especially the acquisition of computer hardware and software components needed to develop the in-house software components.

8.3.1 Inputs to the Implementation Phase

- System Design Specifications,
- System Acceptance Criteria,
- System Test Plans,
- System Implementation Plans.

8.3.2 Activities of the Implementation Phase

- Prepare Procurement Specifications;
- Procure and Install Hardware and Software;
 - Prepare, Publicize, and Issue the Request for Proposal (RFP) and Technical Specifications,
 - Survey the quality and configuration control operations of prospective vendors,
 - Evaluate proposals,
 - Negotiate contract with the selected vendor,
 - Qualify the vendor product,

- Install the CI in the facility, and
- Conduct CI acceptance tests;
- Develop Required Software Products;
- Develop Operation Plans;
- Develop Operational Procedures;
- Develop Training Materials;
- Define Facility Specifications;
- Select Suitable Site; and
- Prepare Selected Site.

8.3.3 Outputs of the Implementation Phase

- Maintenance and User Manuals,
- Operational Procedures, and
- Training Materials.

8.4 INTEGRATION AND TEST

The objective of this phase is to prove to the client that the integrated system meets all the requirements specified in the system specification.

8.4.1 Inputs to the Integration and Test Phase

- System specification and
- System test Plans.

8.4.2 Activities of the Integration and Test Phase

- Prepare system test specifications,
- Prepare system test procedures,
- Test system component,
- Integrate components and test, and
- Run acceptance tests.

8.4.3 Outputs of the Integration and Test Phase

- An accepted system

8.5 TURNOVER

CSC describes Turnover as a series of activities that may continue throughout the development process. The inputs to these activities include:

- Operational procedures,
- Training materials, and
- User and maintenance manuals.

8.5.1 Activities of the Turnover Phase

- Install accepted system,
- Conduct training,
- Ensure operational readiness,
- Turn over system,
- Operate system in transition mode, and
- Maintain System.

The output of this phase and the overall development process is a fully operational system.

CSC recommends the establishment and maintenance of a strong configuration management system, including a CCB, throughout the operational life of the system and submit hardware, software, data bases, operational procedures, training, and all documentation of the operational system to strict configuration control.

8.6 PROJECT MANAGEMENT

CSC describes the project manager's responsibilities as spanning four vital areas: planning, organizing, monitoring, and controlling. Figure 8-3 presents a list of tasks corresponding to each of these functions.

As part of the planning function, CSC uses a work breakdown structure of nine items which can be modified by the project manager to meet the needs of different sizes and types of projects. These WBS items are:

- Program management,
- Systems engineering,
- Systems test and evaluation,
- Training,
- Special budgets and accounts,
- Prime mission hardware,
- Prime mission software,
- Data, and
- Site activation and operation.

CSC discusses, within the context of the organizing function, possible structures of the project organization, responsibilities of various managers, and staffing of the project.

ORIGINAL PAGE IS
OF POOR QUALITY

**FIGURE 8-3: FUNCTIONS
AND ACTIVITIES OF THE PROJECT MANAGER**

PLAN	ORGANIZE	MONITOR	CONTROL
DEFINE THE WORK <i>Describe the elements of work necessary to develop the required products</i> ESTABLISH A WBS <i>Relate the elements of work to a product-oriented tree structure (e.g., WBS)</i> ESTABLISH THE BUDGET AND SCHEDULE <i>Estimate the budget, schedule, and development dependencies for each work element</i>	DEVELOP A PROJECT ORGANIZATION <i>Determine the best project structure to implement the plan</i> PERFORM DETAIL PLANNING <i>Define work required to be performed for each WBS element by each organizational entity</i> <i>Describe the work packages using cost account plans</i> DOCUMENT THE PROJECT PLAN <i>Formally communicate the overall plan and organization to all project members, the client, and CSC management</i> STAFF THE PROJECT <i>Begin building the project team</i>	REPORT AND ANALYZE SCHEDULE STATUS <i>Relate the schedule status to the planned budget in terms of dollars</i> REPORT AND ANALYZE COST STATUS <i>Relate the cost status to the work accomplished</i> RELATE SCHEDULE STATUS TO COST STATUS <i>Compare the cost status with the schedule status</i> MONITOR TECHNICAL PERFORMANCE <i>Relate product performance parameters to established budgets</i>	EVALUATE CONTRACT PERFORMANCE <i>Provide visibility via reports and reviews</i> QUALITY CONTROL AND QUALITY ASSURANCE <i>Design and build quality into every product</i> CONFIGURATION MANAGEMENT <i>Establish rigorous control over configuration baselines</i> DATA MANAGEMENT <i>Supervise distribution of data for the project</i> CONTRACT MANAGEMENT <i>Ensure contractual matters are appropriately handled</i>

[Reference 1]

8.7 SYSTEM ENGINEERING

The systems engineering activities identified in CSC's system development methodology are summarized as follows:

- System engineering planning activities;
 - Systems engineering management plan,
 - Procurement plan,
 - Facility plan,
 - System integration plan,
 - Acceptance test plan,
 - Material management plan,
 - Training plan,
 - Operations plan,
 - Maintenance plan,
 - Logistics plan, and
 - Installation and turnover plan;
- System reviews and baselines;
 - System requirements review (SRR),
 - System design review (SDR),
 - Software specification review (SSR),
 - Preliminary design review (PDR),

- Critical design review (CDR),
 - Functional/physical configuration audit,
 - Functional baseline,
 - Allocated baseline,
 - Development baseline, and
 - Production baseline;
- Identifying system-level requirements;
 - Analyzing system-level requirements;
 - Documenting, reviewing, and baselining system-level requirements;
 - Developing alternative architectures;
 - Realizing candidate architectures;
 - Analyzing candidate systems;
 - Selecting the system;
 - Establishing the system design baseline;
 - Developing specifications for acquisition items; and
 - Providing systems engineering support during the system implementation phase.

8.8 SUMMARY

CSC's methodology for system development has been developed primarily for computer systems consisting of both hardware and software. While the methodology seems sufficiently flexible to be applicable to other types of systems and projects of varying sizes, the adaptation must be made by the project manager or the systems engineer, and no guidelines have been established for making such adjustments.

The CSC development model is sequential, a water-fall type model, with some overlapping of the phases. This overlapping allows for more-effective utilization of resources. The methodology includes a mechanism for controlling changes, but stresses careful definition of the requirements in the early stages of development to avoid extensive changes later on. Although CSC reviews and validates the requirements, the customer is generally considered to be the entity with knowledge of the requirements. The DSDM control change mechanism is developed to accommodate changes that are within the scope of the contract.

REFERENCE

1. Steppel, S., et al, Digital System Development Methodology, Version 2.0, Computer Sciences Corporation, 1985.

9. REVIEW OF SYSTEMS ENGINEERING FOR INTEGRATED HARDWARE/SOFTWARE APPLICATIONS

Integrated Computer Systems (ICS) conducts training programs in systems engineering methodology. The methodology which is presented has a life cycle which includes:

- Conceptual Design;
- System Requirements Analysis;
- System Design;
- Subsystem Implementation; and
- System Integration, Validation, Delivery, O and M, and Disposal.

9.1 CONCEPTUAL DESIGN

9.1.1 Inputs to the Concept Design Phase

- Program, project, mission, and product requirements and constraints,
- Budget allocation,
- Defined period of performance, and
- Project definition and sponsor expectations.

9.1.2 Activities of the Concept Design Phase

- Defining the product context:
 - Customer objectives,
 - Project requirements,
 - System requirement,
 - Project policies,
 - Constraints,
 - External requirements, and
 - Project requirements review;
- Developing the conceptual design:
 - Defining and evaluating options,
 - Evaluating inheritance,
 - Evaluating make or buy decision,
 - Performing trade studies, and
 - Understanding the technology issues:
 - State-of-the-Art technology,
 - Off-the-shelf designs,
 - Technology forecasting; and
- Defining the conceptual baseline.

9.1.3 The Systems Engineer's Role in the Concept Design Phase

- Planning;
- Team organizing:
 - Identify skill needed vs time,
 - Define team operating plan; and
- Team leading:
 - Conducting meetings,
 - Reporting (upward and downward),
 - Managing the interdisciplinary team mechanics,
 - Understanding the product context.

9.1.4 Outputs of the Concept Design Phase

- Baseline conceptual design,
- Plan for follow-on studies,
- Identified technology drivers, and
- Study report.

9.1.5 Exit Criteria for the Concept Design Phase

A sound baseline conceptual system design and successful completion of the Conceptual Baseline Design Review.

9.2 SYSTEM REQUIREMENTS ANALYSIS

9.2.1 Inputs to the System Requirements Phase

- Baseline conceptual design,
- Plan for follow-on studies, and
- Identified technology drivers.

9.2.2 Activities of the System Requirements Phase

- Developing systems requirements;
- Requirements development:
 - Generating system requirements,
 - Documenting system requirements,
 - Organizing requirements,
 - Evaluating and validating requirements, and
 - Designing;
- Requirements flowdown:
 - Selecting criteria and partitioning system,
 - Conducting Preliminary Design Review--system level,
 - Defining Functional Baseline,

- Allocating requirements to subsystems,
- Defining the Allocated Baseline,
- Conducting Preliminary Design Review--subsystem level;
- Completing the Allocated Baseline:
 - Documenting the design, and
 - Conducting the Critical Design Review.
 [system requirements "frozen" at this point]

9.2.3 Role of the Systems Engineer in the System Requirements Phase

- Re-staff and lead the interdisciplinary design team,
- Guide the creation of the requirements infrastructure,
- Lead trade studies and requirements analysis activities,
- Lead generation of system requirements,
- Assure traceability of project requirements to system requirements,
- "Flowdown" the requirements and establish necessary baselines,
- Verify that subsystem requirements are responsive to system requirements,
- Verify compliance with incremental development plans, and
- Support project system requirements reviews.

9.2.4 Outputs of the System Requirements Phase

- An organized/prioritized set of system requirements, and
- A system-level baseline design.

9.2.5 Exit Criteria for the System Requirements Phase

- A sound system baseline and
- Adequate completion of the system CDR.

9.3 SYSTEM DESIGN

9.3.1 Inputs to the System Design Phase

A set of agreed-to system and subsystem level functional requirements and constraints (subsystem PDR and system CDR).

9.3.2 Activities of the System Design Phase

- Developing optional designs:
 - Generate a matrix identifying the features and attributes of the design options,
 - Generate a matrix comparing design options with requirements and constraints,
 - Perform design trade studies,
 - Apply the evaluation criteria,
 - Assess the winning design, and
 - Iterate if necessary;
- Verifying system:
 - Identify verification method,
 - Define best level and environment to accomplish verification, and
 - Generate a "system verification requirements" document;
- Establishing the detailed design:
 - Expanding the level of detail, and
 - Responding to change;
- Modeling:
 - Evaluate alternative modeling approaches:
 - Analytical,
 - Simulation, and
 - Prototyping;
 - Develop and apply models.

9.3.3 Role of the Systems Engineer in the System Design Phase

- Assure subsystem compliance with system requirements,
- Lead implementation trade studies,
- Document and control the implementation,
- Support configuration management,
- Restructure and lead interdisciplinary team to work on design tasks,
- Plan design tasks,
- Audit and control technical resource margins,

9.3.4 Outputs of the System Design Phase

- A detail design--down to the subsystem major assembly level:
 - An agreed-to set of subsystem and system requirements,
 - A point design that meets the requirements,
 - An updated estimate of the technical resources required, and
 - A realistic implementation plan.

9.3.5 Exit Criteria for the System Design Phase

Successful completion of and acceptance of the subsystem detailed design (product baseline) as approved at the subsystem critical design review.

9.4 SUBSYSTEM IMPLEMENTATION

9.4.1 Inputs to the Subsystem Implementation Phase

- An approved subsystem design, and
- An approved implementation plan.

9.4.2 Activities of the Subsystem Implementation Phase

- Establishing the product:
 - Developing hardware/software assemblies/modules:
 - Monitor subsystem engineer's designs,
 - Check system-related items,
 - Ensure software design meets requirements and standards, and
 - Ensure balanced modularity to cost ratio;
 - Concurrent hardware/software/procedures implementation; and
 - Auditing product development and resource utilization;
- Verifying the design:
 - Verify the product; and
 - Prepare for system integration:
 - Review and approve system integration test procedures, and
 - Prepare test monitor data sheet.

9.4.3 Role of the Systems Engineer in the Subsystem Implementation Phase

- Participate in creation and selection of the product,
- Participate in product development activities,
- Audit development to ensure product compliance with requirements,
- Participate in resolution of design problems and setbacks,
- Coordinate iteration of product implementation, and
- Verify compliance with verification requirements.

9.4.4 Output of the Subsystem Implementation Phase

An integrated hardware/software system that has been functionally verified.

9.4.5 Exit Criteria for System Implementation Phase

Successful completion of the product subsystem review.

9.5 SYSTEM INTEGRATION, VALIDATION, DELIVERY, O AND M, AND DISPOSAL PHASE (SYSTEM INTEGRATION/DELIVERY PHASE)

9.5.1 Inputs to the System Integration/Delivery Phase

- Subsystem inputs:
 - Completed subsystem elements, and
 - Formal delivery paperwork; and
- System inputs:
 - System verification/acceptance criteria and requirements, and
 - Special test requests/procedures.

9.5.2 Activities of the System Integration/Delivery Phase

- Defining integration and test strategy:
 - Facilities, personnel and tools,
 - Verification requirements handling,
 - Organize verification requirements (system design phase),
 - Ensure verification test plan adequacy, and
 - Trace satisfactory completion of all verification requirements (concurrent with test execution);
- Test procedures:
 - Ensure early definition of adequacy tests environment,
 - Review and approve incremental generation of individual procedures, and
 - Ensure procedures maintained during testing; and
- Operations considerations:
 - Ensure adequate attention to operations requirements, and

- Use verification testing to "test" the user and operations manuals.
- Managing the integration process:
 - Roles of supporting elements, and
 - Coordinating and managing change;
- Integrating and verifying the product:
 - Monitoring the tests:
 - Integrating the hardware/software systems: key issues,
 - Test preparation,
 - Organizing team support,
 - Monitoring test activity,
 - Testing failure tolerance,
 - Post-test activities,
 - Verifying incremental deliverables, and
 - Alpha and beta site testing;
 - System delivery reviews:
 - User acceptance test
- Coordinating the delivery and user support:
 - Deliver the product,
 - Release documentation,
 - User training support,
 - Failure recovery and error correction,
 - Sustaining engineering, and
 - System phaseout.

9.5.3 Role of the Systems Engineer in the System Integration/Delivery Phase

- Modify and track completion of verification,
- Provide system expertise for detailed test planning,
- Review and concur on detailed procedures,
- Provide system expertise to test execution and verification of results,
- Identify and lead resolution of system-level problems,
- Coordinate change activity, and
- Define system deliverables.

9.5.4 Outputs of the System Integration/Delivery Phase

- Summary test and status reports,
- A completed verification traceability matrix,
- System functional or performance liens, idiosyncrasies, and
- System installation and training support activities.

9.5.5 Exit Criteria for the System Integration/Delivery Phase

- Verification of delivery:
 - Completion of verification,
 - Final acceptance review, and
 - Product packaged and documented per contract agreement; and
- Initial operations documentation, support in place:
 - Facilities,
 - Software,
 - Procedures,
 - Operations personnel, and
 - Logistics for maintenance in place.

9.6 GENERIC LIFE CYCLE ACTIVITIES

- System life cycle reviews;
- Configuration management;
- Test considerations;
- Incremental development;
- Manufacturability considerations;
- Life cycle costing; and
- System size considerations.

9.7 SUMMARY

The methodology presented by ICS consists of five phases. It provides considerable details on seven generic activities that are conducted in each phase. It provides even greater detail on tasks that are specific to each phase. The presentation structure [Input to the phase, Output of the phase, Exit Criteria, Role of the System Engineer, activities, together with examples and applicable tools and techniques] makes it fairly easy to follow and understand.

The methodology does not address the planning and management of the project or the planning and management of the systems engineering functions in detail; however, tailoring the methodology to the type and size of project is a task of the systems engineer and is defined in the Systems Engineering Management Plan. The development phases of the methodology are sequential--a "water fall" type approach, and while the document discusses iteration, it is primarily iteration within the phase of development that is incomplete. The methodology does not address the process of procuring either skills or other resources.

The development phase of the methodology is sequential, and the methodology calls for the reorganization of the project team at the completion of each major phase. This raises concerns about

flexibility with regard to managing the human resources. The methodology seems to advocate a matrix structure for the management and development of the project, but does not address the issues of identifying, selecting, and acquiring the best human resources and addresses development as the only option for hardware and software acquisition.

While the methodology identifies the type of document that is required at different points in the development process, it provides little or no information on the structure or the data items to be included in such documents. It should be noted that documentation is not considered as one of the Generic activities of the development phases. The interdisciplinary team seems to be the only mechanism for managing communication within the project.

While several reviews are conducted through the system development phases, it appears that there are no mechanisms to determine the extent to which the methodology is being followed.

REFERENCE

1. The Learning Tree, Course 348: Systems Engineering for Integrated Hardware/Software Applications, (Vienna, Virginia: Integrated Computer Systems Publishing Company, Inc.), 1988.

10. REVIEW OF JET PROPULSION LABORATORY SYSTEMS DEVELOPMENT METHODOLOGY

Jet Propulsion Laboratory's (JPL's) methodology for management of systems development (D-5000) is designed for large complex systems, with many levels of development; small systems, with only one or two levels of development; single level systems; systems that involve coordinated disciplines; systems implemented in software only or systems implemented in hardware, software, and people/procedures; developer operated systems as well as user operated systems; in-house as well as contracted systems.

The documentation consists of JPL System Development Management Package (D-5000); seven handbooks: Systems Management, Systems Engineering, Test Engineering, Operations Engineering, Software Engineering, Quality Assurance, and Configuration Management; and a Catalog of Services offered Systems, Software, and Operations Resource Center (SSORCE)--a systems development support organization at JPL.

10.1 JPL's PRINCIPLES OF SYSTEM DEVELOPMENT

10.1.1 Separation of Requirements from Design

- System Development Specification (requirements) should describe what is to be done, not how it should be done;
- System Development Specification (design and plans) specify how requirements should be done;
- D-5000 discourages embedding of design in requirements, when there are requirement which intentionally constrains the design options, they can be listed separately from requirements under a heading "design constraints";

10.1.2 Meaning of Phase Exit and of Baselines

- Phase exit corresponds to a review board decision that (1) the information in the System Development Specification (SDS) has reached the Baseline specified for that phase, and that (2) the developing organization is ready to begin executing the actions defined for the next phase;
- Phases are named for the dominant activity that occurs in each phase. This naming strategy does not imply that activities such as planning, requirements, and design are constrained to occur in only one phase. In fact, D-5000 assumes the opposite

is true: that planning, requirements, and design milestones and activities occur in every phase.

- The concept of phases in the D-5000 model refers to the maturity and degree of detail of the data items--an iterative process. This is different from the waterfall model which generates new sets of data items in phase--a sequential process;

10.1.3 Early Involvement of Test and Operations

- D-5000 specifies full Test and Operations Engineering participation during the early development phases;
- Test and operations resource requirements (i.e., budgets and schedules) should be understood before approving plans;
- Requirements addressing system testability and operability should be defined before approving requirements;
- Cost and schedule impacts of testing system requirements should be evaluated at the same time as implementation costs during the requirements scrubbing process;
- System design trade studies should be evaluated in terms of life cycle costs (e.g., including operations costs), not just development costs;

10.1.4 Separation between Architectural and Implementation Design

- D-5000 advocates that architectural design should be functional, that is, "designed or developed chiefly from the point of view of use;"[Webster]
- D-5000 advocates that implementation design should be physical, that is; "of, relating to, or according with material things;"[Webster]
- Architectural design that are implementation dependent should be avoided. Rather, architectural design should permit analysis and selection of an implementation design from among many viable alternatives;
- This principle of separating architectural design from implementation design is particularly important for systems that are expected to change with evolving technology;

10.1.5 Functional Partitioning

- D-5000 advocates that higher level systems be partitioned into subsystems by function, not by software, hardware, and/or operations;
- D-5000 discourages premature assignment of functions to hardware, software, people, and procedures;
- D-5000 advocates performing implementation media trade studies, analysis, and decisions as part of lower-level design, where the appropriate expertise resides;

10.1.6 Alternative Designs/Trade Studies

- The D-5000 model specifies that the design process should generate more than one response to requirements. The design process should generate alternative designs, develop and apply selection criteria, and perform trade studies; and
- The D-5000 model treats plans similarly.

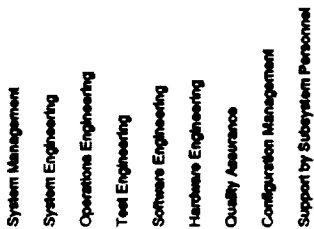
10.2 SYSTEM DEVELOPMENT SPECIFICATION

D-5000 sorts the system development specification into three sub-sets: process specification, system specification, and certification specification as illustrated in Figure 10-1. Because the requirements which form the backbone of the specification become apparent as the system develops, developing the specification is treated as an evolving process which ends at the Operations Certification Phase.

10.2.1 Process Specification

The process specification contains the management requirements and the management plan. It reaches significant maturity during Phase One, but must be updated during each of the succeeding phases. The responsibility for the management specification rest with the System Management discipline, although much of the information needed to prepare this specification is provided by the other disciplines. The process specification includes:

- Management plan requirements, and
 - Management context and goals,
 - Control requirements,
 - Negotiated System development resources,
 - Risk management requirements,
 - Reporting and review requirements,
 - Documentation requirements,



- Configuration management requirements,
- Quality assurance requirements, and
- Management standards and constraints;
- Management plan;
 - Management approach waivers,
 - Work breakdown and account structure,
 - Organization and staffing,
 - Risk management,
 - Reporting and review,
 - Documentation,
 - Configuration management,
 - Process quality assurance,
 - Schedules and budgets,
 - Development environment, training, and tools,
 - Discipline management plans,
 - Detailed phase plans, and
 - Subsystem management requirements;

10.2.2 System Specification

The system specification contains the system requirements and the system design. It is developed in Phase One, only to the extent needed to support planning.

The system requirements are developed significantly in Phase Two, and updated in each phase thereafter. The system design is developed in Phase Two, only to the extent needed to support the development of the system requirements and certification requirements. The major development of the system design occurs in Phase Three, and is updated thereafter.

The responsibility for system specification rests with the System Engineering discipline, although information is required from the other disciplines. The contents of the System Specification include:

- System requirements and constraints, and
 - System goals,
 - Operational concept,
 - Testability requirements,
 - Operability requirements,
 - System context/interfaces,
 - Functional requirements,
 - Performance requirements,
 - Constraints on system design, and
 - Attributes;
- System design:
 - Design approach,
 - Architectural design,

- Mode design,
- Allocation of system resources to subsystem,
- Margin management plan,
- Interface specification,
- User's guide,
- Maintenance guide,
- System operation plan,
- System software plan,
- Design constraints on subsystems, and
- Subsystem requirements; and

10.2.3 Certification Specification

The certification specification consists of the system certification requirements and the system certification design. Like the system specification, it is developed in Phase One, only to the extent needed to support planning.

Consistent with the process of developing the system specification, the certification requirements are developed extensively in Phase Two and the certification design is developed mainly in the Design Phase, Phase Three. Both certification requirements and certification design are updated in all succeeding phases.

The System Engineering discipline also has responsibility for the certification specification. The certification specification consists of:

- Certification requirements:
 - Certification goals,
 - Phased certification requirements,
 - Test requirements,
 - Demonstration requirements,
 - Inspection requirements,
 - Modeling/analysis requirements, and
 - System quality assurance requirements; and
- Certification design.
 - (details being developed by JPL)

10.3 SYSTEM DEVELOPMENT EXECUTION

D-5000 defines system development execution in terms of a seven-phase life cycle.

- Planning,
- Requirements,
- Design,
- Implementation,
- Integration and Test,

- Environmental Integration, and
- Operations Certification.

Within each phase, D-5000 requires one iteration through the SDS, documenting the SDS maturity (more details and new requirements), the degree of detail required for making the decision to begin the next phase, and actions that must be taken in that phase to develop the new information to be added to the evolving SDS Baseline.

JPL recognizes the hierarchical nature of systems, and has developed a methodology that can be implemented at any level, by specifying the deliverables and receivables between any level and its subordinate levels. The organization of the life cycle phase for three levels of the system development structure is presented in Figure 10-2.

D-5000 has defined seven technical disciplines to which development task and deliverables may be assigned. A staffing plan required in Phase One addresses the assignment of staff to these disciplines, depending on the nature and size of the developmental project. These disciplines are:

- System Management,
- System Engineering,
- Test Engineering,
- Operations Engineering,
- Software Engineering,
- Quality Assurance, and
- Configuration Management.

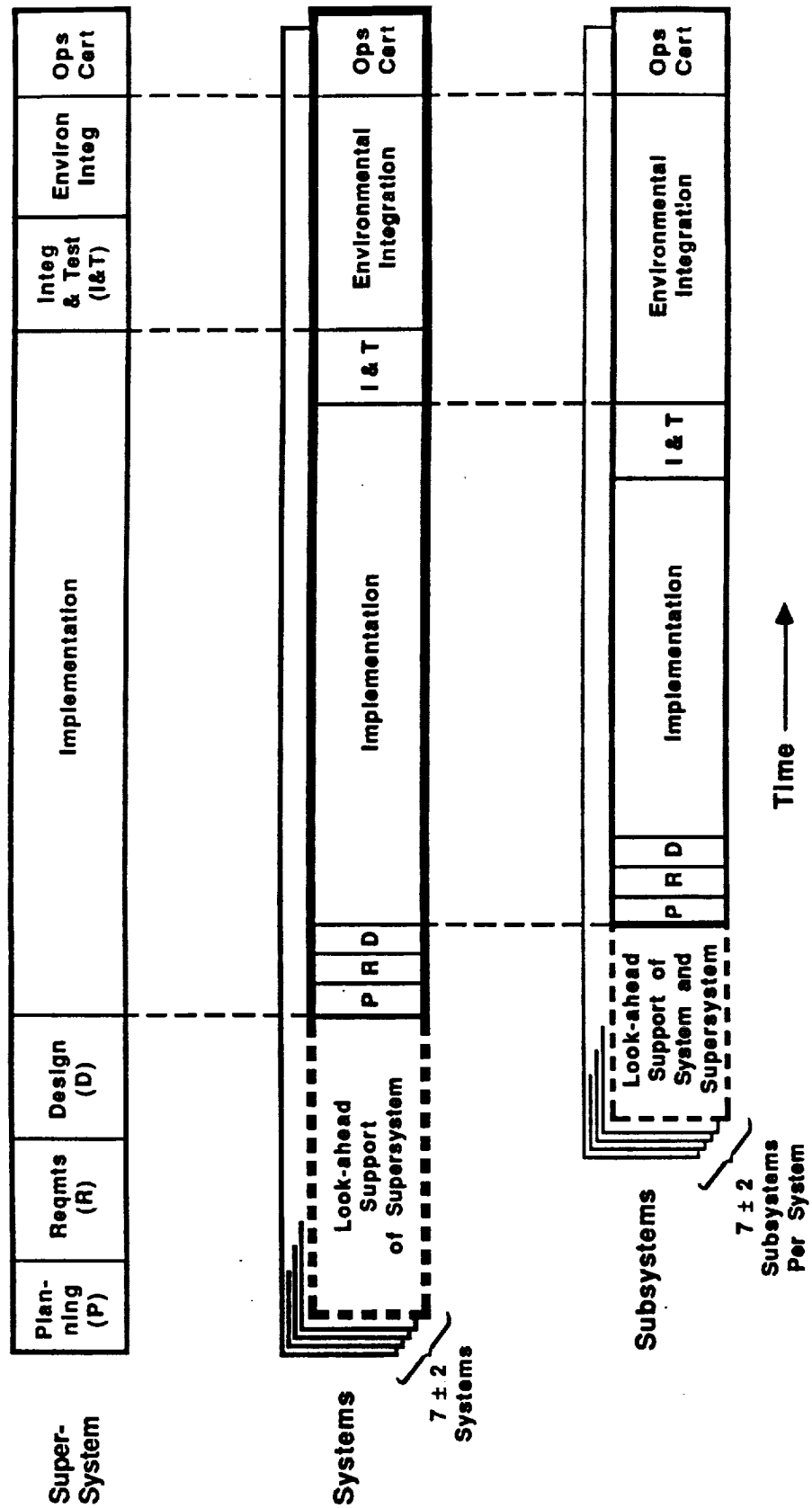
10.4 SUMMARY

The JPL system engineering methodology is being developed. The System Development Management Guide documents the developments to date. Section 4 (30 pages) which describes the methodology as it relates to specifications is completed, but Section 5 which detail the execution aspects has not been released.

While the methodology, in its current state of development, does not contain a detailed set of tasks, it does provide a very detailed set of data items to be generated. The handbooks being prepared for the different disciplines is expected to provide further details on such tasks.

The hierarchical framework of the methodology makes it very adaptable to different projects of different sizes. Large projects (super-systems) can be decomposed in smaller systems, subsystems, and sub-subsystems, with a 7+2 Rule being used structure the decomposition process. Rules have been developed for level-to-level contracting with disciplines within or external to JPL. The current documentation does not describe the procurement process.

FIGURE 10-2: SYSTEM DEVELOPMENT PHASES AT SUCCESSIVE SYSTEM LEVELS



[Reference 2]

The Looking Ahead Support at the front-end of the development process for all but the highest level of the structure [Figure 2] allows for inputs from the technical disciplines in the Planning, Requirements, and Design phases of the higher level in the structure.

The iterative approach coupled with the principle that plans, requirements and designs should be constantly evolving, make incorporation of new information and requirements a very natural process in this methodology.

The methodology identifies seven technical disciplines to which WBS items can be assigned. This assignment of action and data items to disciplines makes the methodology very independent of organizational structure, and to some extent specifies the types of skills required to perform required tasks and process required data items of different types.

REFERENCES

1. SSORCE, The JPL System Development Management Guide: Working Draft, (Pasadena, California: Jet Propulsion Laboratory, California Institute of Technology), February 1989.
2. Ruskin, Arnold M., Carraway, John B., and Singer, Michael J., "Developing one-of-a-kind Systems Systematically", (Pasadena, California: Jet Propulsion Laboratory, California Institute of Technology), to be presented at the Second International Conference on Engineering Management, Toronto, Canada, September 1989.

11. REVIEW OF DEFENSE SYSTEMS MANAGEMENT COLLEGE SYSTEMS ENGINEERING MANAGEMENT METHODOLOGY

The Defense Systems Management College's (DSMC) systems engineering methodology has been designed for engineering development programs within the defense establishment which are characterized as follows:

- Large design teams are needed,
- Designers are highly specialized,
- Many contractors are involved,
- Contractors are located throughout the country, making communication complicated,
- Many related hardware and software systems are under concurrent development,
- Operational and logistics support requirements are very complex,
- Development time is severely constrained, and
- A high level of advanced technology is inherent in many subsystems.

The methodology used by the DoD governs major and non-major defense acquisition programs under the command of the Secretary of Defense. These instructions are applicable to the Office of the Secretary of Defense, the Military Departments, the Organization of The Joint Chiefs of Staff, the Unified and Specified Commands, and the Defense Agencies.

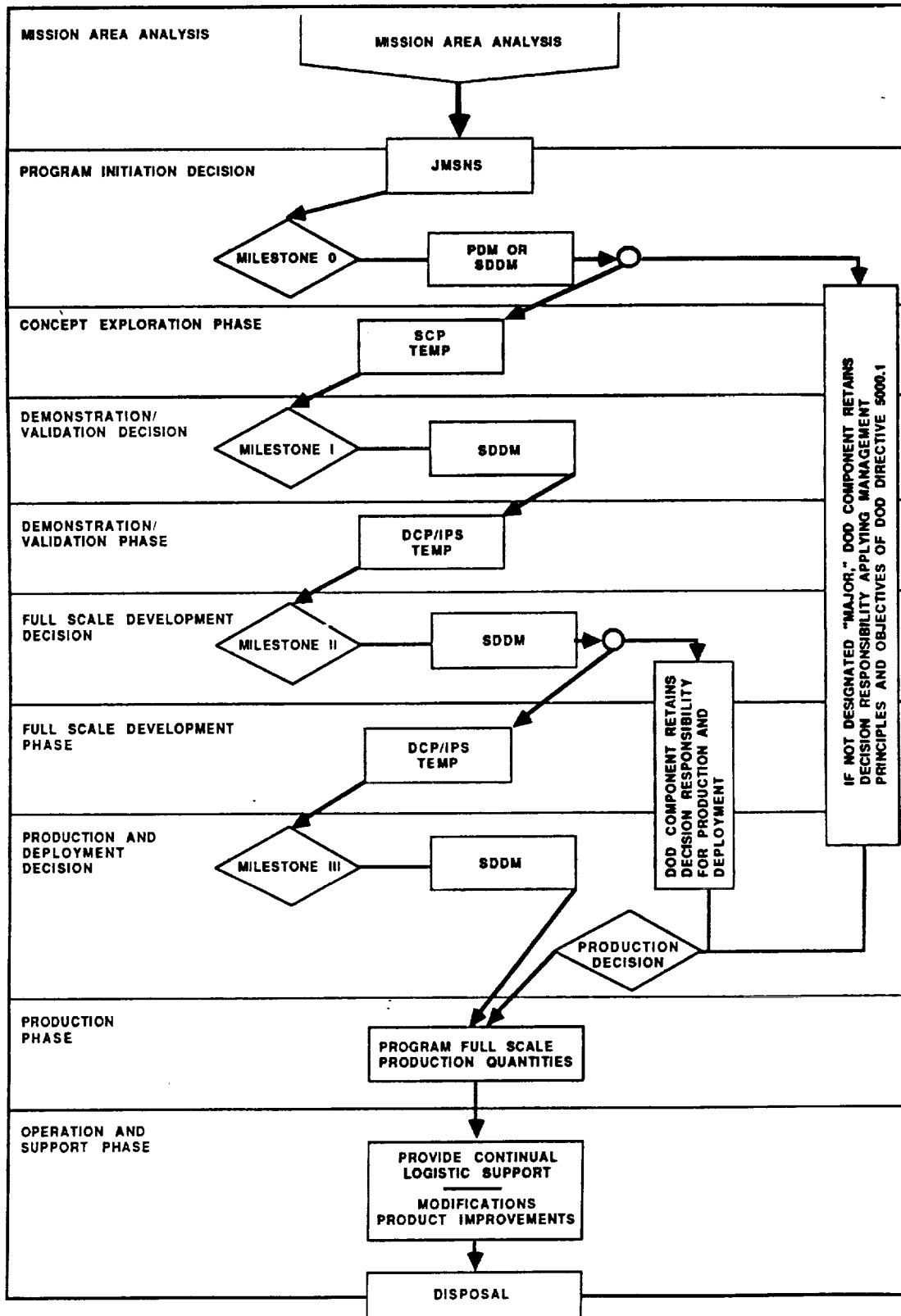
The Department of Defense Acquisition System is a single uniform system whereby all equipment, facilities, and services are planned, designed, developed, acquired, maintained, and disposed of within the Department of Defense. The system involves establishing policies and practices that govern acquisitions, determining and prioritizing resource requirements, directing and controlling the process, contracting, and reporting to Congress.

11.1 THE LIFE CYCLE PHASES

The acquisition process is initiated by an on-going activity, entitled "Mission Area Analysis" and proceeds through the following five phases, with appropriate decision points within. See Figure 11-1.

- Concept Exploration,
- Demonstration/Validation,
- Full Scale Development,
- Production, and
- Operations and Support.

FIGURE 11-1: DoD LIFE CYCLE PHASES AND MILESTONES



[Reference 1]

Mission area analysis is an on-going activity for identifying deficiencies in existing defense capabilities or determining more effective means of performing assigned tasks within assigned mission areas. When deficiencies or opportunities are identified, various analyses are conducted of the threat, political implications, and alternatives to achieving the new capability, including: new development, redeployment of an existing military resource, use of commercial systems, and tactical changes. When no other alternative is available, the end product of this activity is the Justification for Major System New Start (JMSNS) which defines the mission need, identifies constraints, and outlines the initial acquisition strategy.

11.1.1 Concept Exploration

Upon approval of the JMSNS by the Secretary of Defense, the program office identifies all reasonable system alternatives that may satisfy the mission need. The program manager selects for further development those that meet cost, risk, schedule, and readiness objectives.

Alternative system design concepts are explored through competitive, parallel, short-term contracts; alternative methods of logistic support are examined through logistic support analysis; and producibility is analyzed through producibility engineering and planning. Contractors are provided with operational employment intentions, mission performance criteria, and life cycle cost estimating factors.

The industry's systems engineering activity during this period is based on system requirements provided with the statement of work (SOW). These requirements are translated into alternative design concepts, through functional analysis, synthesis, and trade-off analysis. For each segment of the design concept, allocated requirements, interface identifications, and technical budgets are produced as systems engineering products.

The system descriptions, and associated risks, cost, and development time estimates are used by the government to establish a functional baseline, usually in the form of a Type A specification (MIL-STD-490A). The System Engineering Management Plans (SEMPs), Integrated Logistic Support Plans (ILSPs), Test and Evaluation Master Plans (TEMPs), and other functional plans are normally initiated during this phase. A SRR is accomplished to determine the extent to which selected contractor design concepts satisfy the stated mission need.

11.1.2 Demonstration/Validation (D/V) Phase

Upon approval of the required documentation of Phase One, the D/V contractor is selected. At this point systems engineering becomes a contractor effort, often by two or more contractors. The objective of the D/V phase is to identify and analyze major system alternatives, examine risky subsystems, and determine whether or not to proceed with full scale development. The product of this phase is normally the allocated baseline (or design requirements), a set of firm and realistic subsystem performance specifications that meets the operational and support requirements. This baseline also incorporates technological approaches developed to satisfy requirements established at the system level by the functional baseline.

Another major product of the D/V phase is the System Engineering Management Plan, which includes plans for risk alleviation and identifies the schedule for producing all required plans for the supporting engineering specialties, such as electromagnetic compatibility/electromagnetic interference, reliability, maintainability, safety, integrated logistic support, and configuration management.

As the systems engineering progresses from functional to allocated baselines, required configuration items are identified. Elements of the proposed system are continually assessed to identify areas of technical uncertainty that must be resolved in later program phases (risk management). Critical components may be prototyped to reduce risk. A SDR is held at the end of the D/V phase or early in the Full Scale Development phase to review the preliminary allocation of requirements to hardware CIs, data CIs, software CIs, personnel, and facilities.

11.1.3 Full Scale Development Phase

The purpose of the Full Scale Development (FSD) phase is to provide the design documentation necessary to go to full rate production and the ILS documentation necessary to field and fully support the system. This phase begins with the DoD selection of one or more development contractors.

The SEMP and other plans developed in the D/V phase are implemented in this phase. Tests plans are developed, tests are conducted, and test data are audited and compiled.

The SDR is followed by the Software Specification Review of computer software CIs and the Preliminary Design Review. All of which are conducted prior to the start of the detail design. A CDR is conducted of each CI before the design is released for production. Systems engineering activities change considerably after CDR and consists primarily of resolving interface

compatibility problems and solving technical problems discovered during the development testing.

Functional Configuration Audit (FCA) is conducted on each CI before Milestone III. Physical Configuration Audit (PCA) is also conducted on each CI. The PCA may be accomplished during the FSD phase or at the beginning of the Production phase.

The output of the FSD phase is a tested design that meets contract requirements and the documentation necessary to enter the Production and the Operation and Support phases, including product, process, and material specifications; Production Plans; ILSPs; and an RFP for the Production phase.

11.1.4 Production

The primary objective of the production phase is to produce and deliver an effective, fully supported system at an optimal cost. In a production run where many items are to be delivered, manufacturing is usually accomplished in two segments. The first segment starts with low-rate production of initial product batches or blocks. During the second segment, the rate increases to peak rate production as necessary changes resulting from initial operational use, experience, review, audits, testing, and production experience are incorporated.

11.1.5 Operation and Support Phase

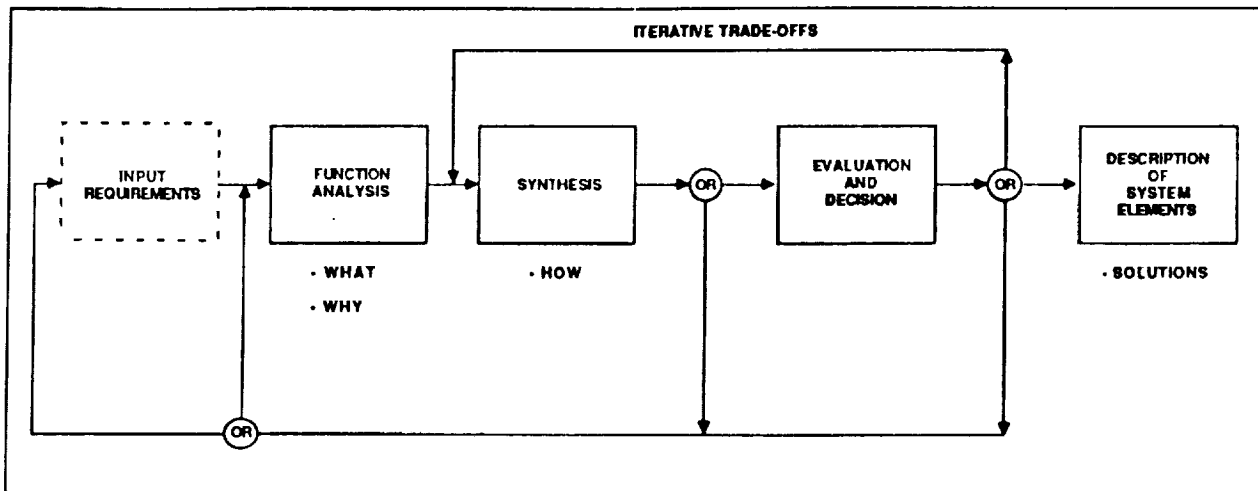
The Operation and Support phase starts with the deployment of the system and continues until disposal (which marks the end of the system life-cycle). The major activities during this period include introducing modifications and product improvements as necessary throughout deployment as well as supporting the fielded system with items such as tools, spare parts, and technical documents.

11.2 THE SYSTEMS ENGINEERING PROCESS

DSMC sees systems engineering as an iterative process, involving three major activities, in which the product element descriptions (the documentation) become more detailed with each iteration, and the final product is production-ready documentation of all system elements. Figure 11-2 illustrates the process. The systems engineering activities are listed as follows:

- Function analysis:
 - Functional identification, and
 - Requirement allocation;

FIGURE 11-2: DSMC SYSTEMS ENGINEERING PROCESS



[Reference 1]

- **System Synthesis:**
 - Schematic block diagrams,
 - Physical modeling, and
 - Mathematical modeling;
- **Evaluation and Decision: Trade Studies:**
 - Trade-off analysis,
 - Trade studies, and
 - Risk template: trade studies; and
- **Documentation.**

Other systems engineering functions include:

- **System definition and control:**
 - Work breakdown,
 - Specification development,
 - Configuration management, and
 - Technical reviews and audits;
- **System performance measurement:**
 - Testing and evaluation, and
 - Technical performance measurement; and
- **Supporting the transition from development to production:**
 - Risk analysis and management,
 - Modification management,
 - Life cycle cost analysis, and
 - Manufacturing and producibility planning.

The System Engineering Management Plan describes the management of all engineering activities in the development process, including the integration of the following technical disciplines:

- Technical performance measurement,
- Producibility,
- Maintainability,
- Quality,
- Human Engineering,
- Safety,
- Logistic support analysis,
- Reliability,
- Production engineering,
- Contamination and corrosion control,
- Parts, materials, and process control,
- Electromagnetic control,
- Nuclear hardening,
- Vulnerability/survivability,
- Weight control,
- Mass properties control, and
- Packaging, handling, storage, and transportation.

11.3 SUMMARY

The Systems Engineering Management Guide is one of a family of educational guides written for the Department of Defense. The Guide has been prepared for program and project management personnel.

The document presents the life cycle phases and the major decision points in the system acquisition process. It presents extensive details on the systems engineering process, including the activities, documentation, review applicable to the different phases, and tools and techniques applicable to the different systems engineering functions. A special application of the methodology to software development is presented in Chapter 20. In addition, the document discusses Modification Management, and the interaction between Integrated Logistic Support (ILS) and systems engineering.

The systems engineering process is described as iterative, but the life cycle development process is strictly sequential, often with different contractors being responsible for the different phases. Thus, iteration may occur within a specified phase, not through multiple phases.

The methodology has been developed for acquisition through a procurement management process, and, as such, acquisition through in-house development is not considered.

In addition to identifying the technical disciplines that are usually required for most significant systems development project, the methodology discusses how the composition of the development team is likely to change through the development phases.

REFERENCE

1. Systems Engineering Management Guide, Second Edition, (Fort Belvoir, Virginia: Defense Systems Management College), December 1986.

12. REVIEW OF SYSTEM ENGINEERING AND ANALYSIS

This systems engineering methodology by Blanchard and Fabrycky (B&F) is intended for the classroom, either the graduate or undergraduate level or for practicing professionals in either industry, business, or government. They have partitioned systems into two categories, natural and man-made, and constrained the application of the methodology to the man-made systems.

The text discusses systems concepts, then presents a customer-to-customer life cycle of seven phases as follows:

- Identification of need,
- System planning,
- System research,
- System design,
- Production and/or construction,
- System evaluation, and
- System use and logistic support.

12.1 SYSTEMS ENGINEERING FUNCTIONS

B&F describe the system engineering function within each of the phases below, and show the feed back nature of the systems engineering process in Figure 12-1.

12.1.1 The Planning Function

The system engineering function often includes marketing and marketing analysis, the performance of feasibility studies, and advanced planning. The engineering involvement in this task depends, to a great extent, on the nature and size of the project.

The performance of feasibility studies involves such areas as needs analysis, identification of possible solutions to meet the need, the screening of alternatives, selection of preferred approaches, and the preparation of proposal. In this area, engineers are initially involved in assessing system performance characteristics and determining the various technical approaches that are possible in responding to the need. Also, given two or more alternatives, the engineer must evaluate each on the basis of technical performance, reliability and maintainability, size and weight, cost, and the like. The engineer will also assist in the preparation of follow-up proposals and/or reports by providing the necessary technical inputs.

The engineer's input to the advanced planning involves assisting in the preparation of product specifications and ensuring

that the various plans and schedules are realistic and technically feasible.

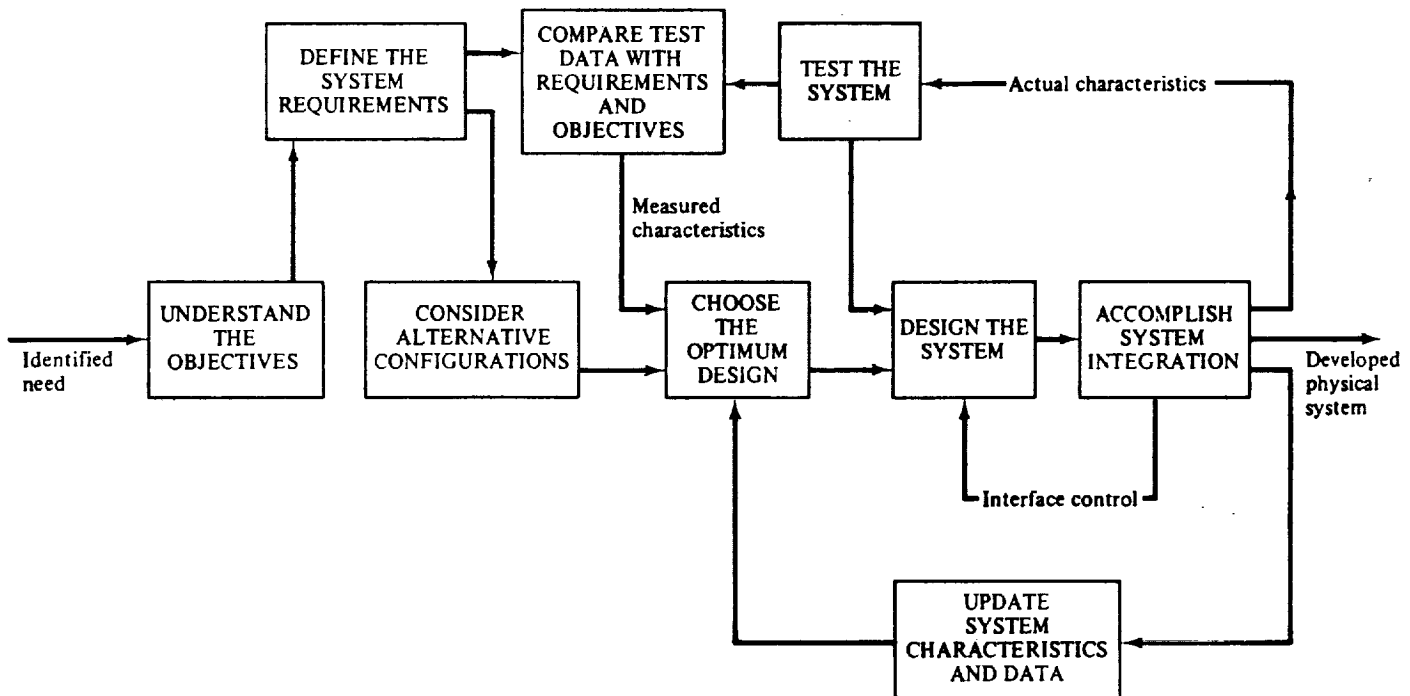
12.1.2 The System Research Function

The research function is subdivided into basic and applied research. In either case, highly specialized engineers working with scientists in scientific fields are required when the objective constitutes advanced knowledge. As research results are generated, it is necessary to convert the new knowledge gained to a form that can lead to system development, production, and ultimate consumer use.

12.1.3 The System Design Function

The design process follows from a set of stated requirements for a given system and evolves through (1) conceptual design, (2) preliminary design, and (3) detail design. The engineer's role in this system phase involves a variety of functions, such as:

FIGURE 12-1: FEEDBACK IN THE SYSTEMS ENGINEERING PROCESS



[Reference 1]

12.1.3.1 The Conceptual Design

- Need and feasibility analysis;
 - System operational requirements:
 - Mission definition,
 - Performance and physical parameters,
 - Operational deployment,
 - Operational life cycle,
 - Utilization requirements,
 - Effectiveness factors, and
 - Environment;
 - System maintenance concept:
 - Organizational maintenance,
 - Intermediate maintenance,
 - Depot maintenance, and
 - Preliminary system analysis;
- Advanced product planning; and
- Conceptual design review.

12.1.3.2 The Preliminary Design

- Functional analysis:
 - Functional flow diagrams,
 - Operational functions, and
 - Maintenance functions;
- Requirements allocation;
- Trade-off and optimization;
- Synthesis and definition; and
- System design review.

12.1.3.3 Detailed design and development

- Detailed design requirements:
 - Design for functional capability or performance,
 - Design for reliability,
 - Design for maintainability,
 - Design for manability,
 - Design for producibility,
 - Design for supportability,
 - Design for economic feasibility, and
 - Design for social acceptability;
- Cost-effectiveness figure of merit;

- Detailed design activities:
 - Establishing the design team,
 - Evolving the detail design, and
 - Applying engineering design aids:
 - Design standards documentation,
 - Design criteria documentation,
 - Computer usage in design, and
 - Development of physical scale models or mock-ups;
- System prototype development:
 - Engineering models;
 - Service test models; and
 - Prototype models;
- Formal design reviews.
 - Equipment design review and
 - Critical design review

12.1.4 Production and/or Construction Function

This phase may constitute (1) the production of a multiple quantity of like items (mass production), (2) the production of small quantities of a wide variety of different items (job-shop type of operation), and (3) the construction of a single item, such as a large structure.

Engineering is directly required in the design and development of a production capability and for defining the resources necessary for a large construction project. The engineering function includes the following:

- Design of facilities for product fabrication, assembly, and test functions;
- Design of manufacturing processes;
- Selection of materials and the determination of inventory requirements;
- Design of special tools, test equipment, transportation and handling equipment, etc;
- Establishment of work methods, time and cost standards for subsequent evaluation of production/construction operations;
- Evaluation of production/construction operations to ensure performance quality, reliability, maintainability, safety, and other desired features.

12.1.5 The System Evaluation Function

Throughout product/system development, it is necessary to perform an evaluation or assessment to ensure that the end result will conform to the initially established requirements and meet the

need(s) of the consumer. The engineer plays a vital role in the evaluation by participating in the activities summarized below:

12.1.5.1 Planning and Requirements for test and Evaluation

- Test and evaluation planning and
- Test and evaluation requirements

12.1.5.2 Test and Evaluation Classification

- Performance tests,
- Environmental qualification,
- Structural tests,
- Reliability qualification,
- Maintainability demonstration,
- Support equipment compatibility tests,
- Personnel test and evaluation, and
- Technical data verification.

12.1.5.3 Preparation for System Test and Evaluation

- Selection of test items,
- Test and evaluation procedures,
- Test-site selection,
- Test personnel and training,
- Test facilities and resources,
- Test and support equipment, and
- Test supply and support.

12.1.5.4 Test Performance and Reporting

- Test data requirements,
- Development of a data subsystem,
- System evaluation and corrective action, and
- Test reporting.

12.1.5.5 System Modification

- Initiate system modifications as required to correct deficiencies.

12.1.6 The System Use and Logistic Support Function

This function involves the consumer use of the product/system through its intended life cycle, the incorporation of product or system modifications, the logistic support requirements necessary

to ensure that the product/system is deployed and operationally available when needed, and the ultimate phase-out and disposal of the product/system. The engineering activities during this phase are summarized below.

12.1.6.1 System Support Requirements

- Maintenance planning,
- Supply support,
- Test and support equipment,
- Transportation and handling,
- Personnel and training,
- Facilities,
- Data, and
- Software.

12.1.6.2 Logistic Support in the System Life Cycle

- Logistic support planning:
 - Maintenance plan,
 - Supply support plan,
 - Test and support equipment plan,
 - Personnel and training plan,
 - Facilities plan,
 - Data plan, and
 - Retirement plan;
- Design for logistic support;
- Logistic support in the production phase; and
- Logistic support for operating systems.

12.1.6.3 Measures of Logistic Support

- Supply support measures,
- Test and support equipment measures,
- Organizational measures, and
- Facility measures.

12.1.6.4 Logistic Support Analysis

12.1.6.5 Logistic Support Test and Evaluation

12.2 SYSTEMS ENGINEERING MANAGEMENT

Systems engineering management, as presented by B&F, involves planning, organizing, staffing, monitoring, and controlling the

process of designing, developing, and producing a system that will meet a stated need in an effective and efficient manner. It provides the necessary overview function(s) to ensure that all needed engineering disciplines and related specialties are properly integrated. It ensures that the system being developed contains the proper mix of hardware, software, facilities, personnel, and data. The objective of systems engineering management is to provide the right item, at the right location, at the right time, with minimum expenditure of human and physical resources.

The methodology developed by B&F addresses systems engineering management from the standpoints of:

- Goals and objectives,
- Organization and Staffing, and
- Engineering decision making.

12.3 SUMMARY

This systems engineering test presents a methodology which contains seven life cycle phases, and systems engineering activities that are associated with each of the phases. The text devotes considerable effort to the system design activities and to analytical tools for systems engineering.

While the approach is of the general structure required by the ND, the methodology does not address acquisition through a procurement process, and it merely touches the organizational and staffing issue associated with developing large engineering systems.

In the area of accountability, the methodology does not specify the documents that are needed at different points through the development process, nor does it specify any procedures for verifying that it is being used as intended, nor does it suggest ways of addressing critical considerations and requirements.

The methodology does not address the issue of projecting or predicting future requirement, or designing systems when the requirements are fuzzy. While the text discusses prototyping, its presentation is more from the standpoints of testing and performance verification than from communication. The methodology barely touches on the topic of modification as a means of extending the system's useful life.

REFERENCE

1. Blanchard, Benjamin S., and Fabrycky, Wolter J., Systems Engineering and Analysis, (Englewood Cliffs, New Jersey: Prentice-Hall), 1981.

13. EVALUATING SELECTED METHODOLOGIES

The criteria discussed in Chapter 5 were applied to the seven methodologies [including MO&DSD's SMP] provided by industry and government or selected from academia. Table 13-1 summarizes the results.

While reviewing the following analysis, one should keep in mind that any judgment that a given feature was not contained in the methodology [the response "N - NO"] or that the handling was inadequate [the response "I - Inadequate"] should not be taken out of context. This is purely from the standpoint of the reviewers understanding of the ND's requirements for systems engineering methodologies.

13.1 SCOPE AND STRUCTURE OF THE METHODOLOGY

All of the methodologies reviewed use a phased development approach, with the number of phases in the life cycle ranging from five to eleven. With the exception of the methodologies being used by the ND and the DSMC, all methodologies provided tasks to be conducted and/or very detailed data items to be collected within each phase.

The need to divide the work into "manageable pieces" and more detailed tasks is recognized in the SMP and in the methodology being used by DSMC, but it is assigned as one of the responsibilities of the project or systems engineering manager.

This lack of detail in the SMP was identified as concern of some of the managers interviewed in Phase One of this project.

13.2 FLEXIBILITY

All of the methodologies adapt to size and complexity by having the project manager or the systems engineering manager vary one or more of the following: the number of, phases in the development cycle, elements in the WBS, tasks and subtasks in the different phases of development, reviews, or required documents--horizontal decomposition. They also support hierarchical decomposition of the project into supersystem, system, subsystems and sub-subsystems, and/or functions until a level is reached where the hardware and/or software design is implemented. The methodologies being used by the ND and ICS seem to be limited to two or three levels of decomposition of this type. The individuals surveyed at the ND seem to need more assistance with decomposing large projects, such as: guidelines for tailoring the methodology to different types and sizes of projects or methodologies which have been pre-tailored for different sizes and types of projects.

All of the methodologies incorporate procedures for change management and most allow for iteration among activities within the current phase of development. The methodologies being used by HUGHES, CSC, ICS, JPL, and B&F allow for iteration between adjacent phases and in some cases through multiple phases. In a few cases, the less rigorous definition of phase boundaries also enhances the process of incorporating new information.

The methodologies being used by HUGHES and CSC address acquisition through both procurement and development. The methodologies being used by ND, ICS, JPL, and B&F discuss development only, even though the ND and JPL recognize and use the procurement option. The methodology being used by DSMC relies exclusively on acquisition by contracting, but does not discuss the procurement process.

TABLE 13-1: SUMMARY OF EVALUATION RESULTS

CRITERIA	ND	HUGHES	CSC	ICS	JPL	DSMC	B&F
1. Scope and Structure of the Methodology							
Does the methodology address activities that are likely to involve engineering work such as design, construction, installation, and operation?	Y	Y	Y	Y	Y	Y	Y
Is it structured to handle large-scale or complex systems?	Y	Y	Y	Y	Y	Y	Y
Is it structured to handle at least one component that is extensively hardware?	Y	Y	Y	Y	Y	Y	Y
Is the methodology partitioned into clearly defined and logical phases, processes, activities, or tasks that can be used as a basis for resource allocation and events such as the start or completion of phases that can be used as milestones or decision points?	I	Y	Y	Y	Y	I	Y

YES - Y

NO - N

YES, BUT INADEQUATELY - I

None of the methodologies, with the exception of CSC's, address the subject of effective utilization of project resources. CSC believes that by removing the rigid boundaries between phases it will provide a basis for more effective utilization of resources. The methodologies which permit similar overlapping of, or extensive interaction among, phases, such as HUGHES's and JPL's were also considered to utilize resources more efficiently.

All of the methodologies reviewed consider defining the staffing and resource needs of the project to be part of the responsibility of project/system engineering manager. The methodologies being used by CSC, JPL, DSMC, and B&f discuss briefly some of the issues relating to staffing large systems development projects. The methodology being used by DSMC discusses how the skills of the project team is likely to change during the different phases of the development cycle. HUGHES's methodology discusses

TABLE 13-1: SUMMARY OF EVALUATION RESULTS CONTINUED

CRITERIA	ND	HUGHES	CSC	ICS	JPL	DSMC	B&F
2. Flexibility							
Does the methodology accommodate systems of varying sizes and levels of complexity?	I	I	I	I	I	I	I
Does the methodology address ways of handling new information, feedback, or unforeseen circumstances (Iterative)?	I	Y	Y	Y	Y	I	Y
Does the methodology allow for acquisition through a variety of approaches (procurement, development, etc.)?	I	Y	Y	I	I	I	I
Does the methodology allow maximum flexibility with time-allocation (scheduling) of resources?	N	I	Y	N	I	N	N
Does the methodology address ways of identifying and selecting the best human and material resources to assign or allocate to its various phases of development?	N	Y	I	N	I	I	I
<p>YES - Y NO - N YES, BUT INADEQUATE - I</p>							

in considerable detail the pros and cons of using "project", "loan-in", and "subcontractor" staff for typical systems development projects. It discusses the process of recruiting, developing, and rewarding system engineers.

13.3 ACCOUNTABILITY

The methodology being used by the ND and the methodology developed by B&F identify the titles and, occasionally, the outlines of the documents to be provided at different points in the development process. All of the other methodologies, except JPL's, identify the titles of the documents along with the Military or IEEE standards that are used in the preparation of those documents. JPL's methodology specify the data items that are needed at different points in the development process, but does not suggest how they should be compiled into one or more reports. The preference at JPL is to maintain the data in an interactive database which could be revised and updated as required through the life of the project.

All methodologies have incorporated project review meetings and other techniques for effective communication of project related information. Establishing these techniques and procedures is part on the project manager's responsibility in the systems engineering methodologies reviewed.

All of the methodologies have specified reviews to be conducted at different points in the development process. These reviews are of the progress and quality of the project, and may not certify the program manager's adherence to the systems engineering approach. With the exception of the ND, CSC, and DSMC, none of the methodologies suggest a systems engineering or peer review of the project management or systems engineering plan. Peer reviews of project/systems engineering management plans is being used by AT&T as a means of verifying the integrity of the approach.[1] Although the PMPs of the ND are reviewed and approved, the current review process has been a point of concern of managers within the ND. The methodology being used by the DSMC reviews the Systems Engineering Management Plan of the competing contractors and uses that information in the process of selecting the winning contractor. The information from the SEMP is used in evaluating progress at different points in the development process. CSC uses its Systems Engineering Management Plan to gain approval from Company management and their clients. After which the plan continues to be developed and approved with each review of the program.

The methodologies have also identified their users, class of system, and scope of their intended applications. The methodologies in use by CSC, JPL, and DSMC have addressed risk and critical requirement. The other methodologies do not address that issue.

TABLE 13-1: SUMMARY OF EVALUATION RESULTS CONTINUED

CRITERIA	ND	HUGHES	CSC	ICS	JPL	DSMC	B&F
3. Accountability							
Does the methodology specify the documentation that is appropriate at different points during its application?	I	Y	Y	Y	Y	Y	I
Does the methodology provide strategies and tools for communication and information exchange to ensure that all participants are aware of significant project decisions and have the most up-to-date information on the project status and activities?	Y	Y	Y	Y	Y	Y	Y
Does the methodology specify management procedures to ensure that it has been applied as intended?	I	I	Y	I	I	Y	I
Does the methodology identify its intended users, class of systems, and scope of its intended applications?	Y	Y	Y	Y	Y	Y	Y
Does the methodology address ways of addressing critical considerations such as national security, risk (environmental, evolving technologies), human safety, etc.?	N	N	Y	N	Y	Y	N
YES - Y NO - N YES, BUT INADEQUATE - I							

13.4 DOCUMENTATION OF THE METHODOLOGY

The documentation of the methodologies in use at ND and JPL is presented in considerable less detail than the other methodologies reviewed. While all of the methodologies use graphics to convey some important relationships, CSC, HUGHES, and ICS summarized inputs, outputs, and activities, and, in some cases, roles of the systems engineer during the different phases of development.

13.5 SPECIAL CONSIDERATIONS OF THE ND

While all of the methodologies present and/or discuss an organizational structure, the organizational requirements of the methodologies seem to follow three basic principles: (1) the project manager should have ultimate responsibility for the project, (2) the systems engineering manager should report directly to the project manager, and (3) the systems engineering manager should have responsibility for the engineering/technical integrity of the project.

All of the methodologies assume continuity with respect to staffing of key positions, especially in the project management and the systems engineering areas. Also, review boards and configuration control boards are assumed to operate throughout the completion of the project. With the exception of HUGHES, none of the methodologies address options for retaining such key personnel. The methodology being used by HUGHES addresses career development for the systems engineer.

TABLE 13-1: SUMMARY OF EVALUATION RESULTS CONTINUED

CRITERIA	ND	HUGHES	CSC	ICS	JPL	DSMC	B&F
4. Documentation of the Methodology							
Is the methodology written clearly, precisely, completely, and at a level of detail that is appropriate for its intended users? Is it a good road map?	I	Y	Y	Y	I	Y	Y
YES - Y							
NO - N							
YES, BUT INADEQUATE - I							

Several of the methodologies tie requirements definition to a single phase. They recognize, however, that changes to the defined requirements will occur during the course of system development; and they have incorporated "change control" procedures; and they allow for iterating, usually within a given phase. The methodologies of HUGHES, CSC, and JPL treat requirements as constantly evolving throughout the development cycle, and have developed reviewing and documenting approaches that are more consistent with their requirements development philosophy.

None of the methodologies provide tools and techniques for predicting or projecting future requirements, nor do they address the possibility of designing or developing systems in the absence of specific requirements. HUGHES, CSC, ICS, DSMC and B&F discuss tools and techniques, however, they seem to fall in productivity improvement or system effectiveness measurement categories.

The methodology being used by DSMC discusses Modification Management in considerable detail.

13.6 ANALYSIS

The methodologies reviewed showed significant differences in structure. At one extreme, the SMP has eleven phases and a WBS of eight pre-defined items. At the other extreme, several methodologies had only five phases and no pre-defined work breakdown structure. Some methodologies recommend a hierarchical decomposition of the work into functions and or subsystems and components.

Having provided these basic structures, some methodologies describe the meaning of the phases and the other structural divisions while other methodologies provided far greater details on: inputs to the phases, tasks to be performed in the phases, documents to be developed, criteria for exiting, and outputs of the phases.

These differences in structure and details may be explained in part by differences in application from the DoD which develops supersystems mainly through procurement to the private contractor whose applications can be more varied in the sense that it may choose either in-house development, procurement, or some combination of the two as its means of acquisition. The actual or perceived requirements of the users of these methodologies may also explain some differences, for example, a more mature organization, with more experienced managers, may conceivably require a less detailed methodology. Still however, accommodations must be made for the junior managers and variations in training and experiences of the senior managers.

TABLE 13-1: SUMMARY OF EVALUATION RESULTS CONTINUED

CRITERIA	ND	HUGHES	CSC	ICS	JPL	DSMC	B&F
5. Special Considerations of ND							
Is the methodology fairly independent of organizational structure?	Y	Y	Y	Y	Y	Y	Y
Does the methodology provide for the retention of key personnel throughout the life-cycle?	I	Y	I	I	I	I	I
Does the methodology provide for the incorporation of requirements identified during the system analysis, design, or subsequent phases?	I	Y	Y	I	Y	I	I
Does the methodology provide tools and techniques for predicting or projecting future requirements, through the planning horizon?	N	N	N	N	N	N	N
Does the methodology suggest strategies and techniques for designing and developing systems in the absence of specific requirements?	N	N	N	N	N	N	N
Does the methodology provide tools and techniques (including graphics and prototyping) for communicating among individuals and various organizations or organizational units working on major systems projects?	N	N	N	N	N	N	N
Does the methodology provide tools and techniques for redesigning and making major modifications to extend the useful life of a system in operation?	N	N	N	N	N	Y	N
YES - Y	NO - N		YES, BUT INADEQUATE - I				

All methodologies start generally with some definition of requirements. It seems clear that some take more responsibility for that activity than others. For example, some methodologies treat requirements as inputs to the process which should be verified as opposed to having responsibility for developing and defining such requirements.

There is very little consistency from one methodology to another in terms of the partitioning of the system development cycle into phases or the activities to be performed within these phases.

The methodology being used by DSMC provide an extensive discussion of modification of systems that have been implemented. This is different from the others which plan for the operation, but terminate the systems engineering activities upon delivery or installation and testing of the system.

Systems engineering is presented within the more general context of project management or systems development, where a program or project manager, with ultimate responsibility for the project or system oversees the systems engineering activities.

The methodologies reviewed took a variety of approaches in specifying the tasks of the systems engineer. At one extreme there is an overall description of systems engineering as a work breakdown item. At the other extreme, the role of the systems engineer was described for each phase of the cycle. Somewhere between both extremes is a description of the systems engineering as an iterative set of activities to be conducted at different points in the systems development cycle.

Some methodologies describe systems engineering as comprising a management function, which supervises all technical inputs to the project and a technical function which conducts the systems engineering process or technical activities. Other methodologies seem not to recognize the management function of systems engineering.

All of the methodologies advocate incremental development and testing as opposed to the "big bang", where all functions and capabilities of a complex system are developed concurrently and incorporated simultaneously. This generally involves baselining-establish a minimum capability (functionality) of the system for initial installation.

All of the methodologies recognize the potential for improving the system by iterating, either within phases of the development cycle or through sequential phases of the cycle.

13.7 SUMMARY

The evaluation conducted in Table 13-1 reveals that there are methodologies which can provide assistance with:

- partitioning the project into phases, work breakdown items, tasks, functions, subsystems, and work packages which can be used in planning the project;
- handling new information, feedback, or unforeseen circumstances;
- acquisition by procurement;
- scheduling of project resources;
- identifying and selecting human and material resources;
- specifying the documentation needed at different points in the development process;
- management procedures to ensure that the methodology is being applied as intended;
- ways of addressing critical considerations such as national security, and risk to humans and the environment;
- clearer, more precise, and more complete documentation;
- techniques for retaining key personnel on the project; and
- tools and techniques for redesigning and modifying to add capabilities and extend the useful life of the system.

Needs of the ND that cannot be satisfied by the methodologies reviewed include:

- improving the process of scoping the basic methodology to projects of different types and sizes;
- tools and techniques for predicting or projecting future requirements;
- strategies and techniques for designing and developing systems in the absence of specific requirements; and
- tools and techniques, including graphics and prototyping, for communicating among individuals and organizations working on major systems projects.

Because none of the methodologies reviewed has all of the capabilities required by the ND, the process of developing methodologies that are more suitable for the ND should involve extracting desired features from a variety of sources, integrating, and testing to verify that they will satisfy the requirements.

REFERENCES

1. Notes from presentations made at Howard University by Mr. Pete D'Amato and Dr. George Campbell, both of AT&T, Bell Laboratories, on October 24 and 27, 1988, respectively.

14. TOOLS AND TECHNIQUES

This chapter presents tools and techniques which seem to be applicable to the ND's systems development activities.

14.1 PARTICIPATIVE DESIGN

14.1.1 Problems of Current Design Approaches

- The need for users to assume a new and unfamiliar role,
- Difficulties communicating with colleagues, and
- Differences in expectations between design group members and management.

14.1.2 Benefits of Participative Design

- Creation of new strategies and systems which users like and find efficient, which they can understand, introduce, and manage themselves.
- Users are more likely to get what they need if they are able to contribute to the design task, especially through the analysis of their own needs.
- In most firms experts are a scarce resource and the involvement of users in design frees these experts to take on a broader range of projects. Instead of being 'doers' they become 'advisors', helping users carry out the information collecting and analysis activities that are a time consuming part of most innovation.
- Groups which are passive recipients of major innovation may be afraid and resistant; whereas those who are involved will learn how to cope, exert control and mold the change to fit their own needs, and the needs of their departments and companies

14.1.3 Design Related Organizations

(1) The Design Group

The Design Group should be composed of representatives from the various functions, or major sub-groups with an interest in the change, as well as from several organizational levels. This "diagonal slice" of the organization will increase the opportunity to have considerable knowledge located in the design group. The

team members should be people who are interested in the project, have the time available for the work, are relatively vocal (including some who make good "devil's advocates"), can think broadly and creatively and have good interpersonal skills.

A continuous problem of most design groups is keeping their constituents and other interested groups informed about what they are doing. The writing and distribution of minutes after each meeting can help, as can regular meetings with the steering committee and occasional meetings with other interested groups. Also 'open meetings' of the design team allows for casual inter-change with interested constituents.

(2) The Steering Committee

The Steering Committee should consist of managers who head up the organization under examination as well as managers of interfacing organizations which could affect or be affected by the change. The design task or contract will normally be agreed on between the steering committee and the design group. The steering committee will set the broad framework within which the project will be carried out. This will cover important company policy on certain issues, sensitive decision areas, design boundaries, etc. It should meet regularly with the design group to discuss progress and problems.

(3) The Consultant

Ideally, the consultant should see himself or herself, and be seen by everyone else, as a facilitator: someone who can guide the project and provide helpful advice when this is required but who is in no sense an expert or leader who takes decisions and then persuades others to go along with them. The design group and that part of the organization which its members represent must see themselves as "owning" the project. If they see the consultant as the expert who will tell them what to do every step of the way, then the perception of ownership shifts to the consultant. Consultants who try to assume the role of project manager may find that this has considerable disadvantage for the design group's learning process.... To this end, the organization must learn to manage its own change process. In order for this to happen, the skills and knowledge of how to manage change must be resident in the organization and not in an external consultant.[1]

14.2 PROTOTYPING

Most systems engineering methodologies begin with an identified need, then go through a process of developing

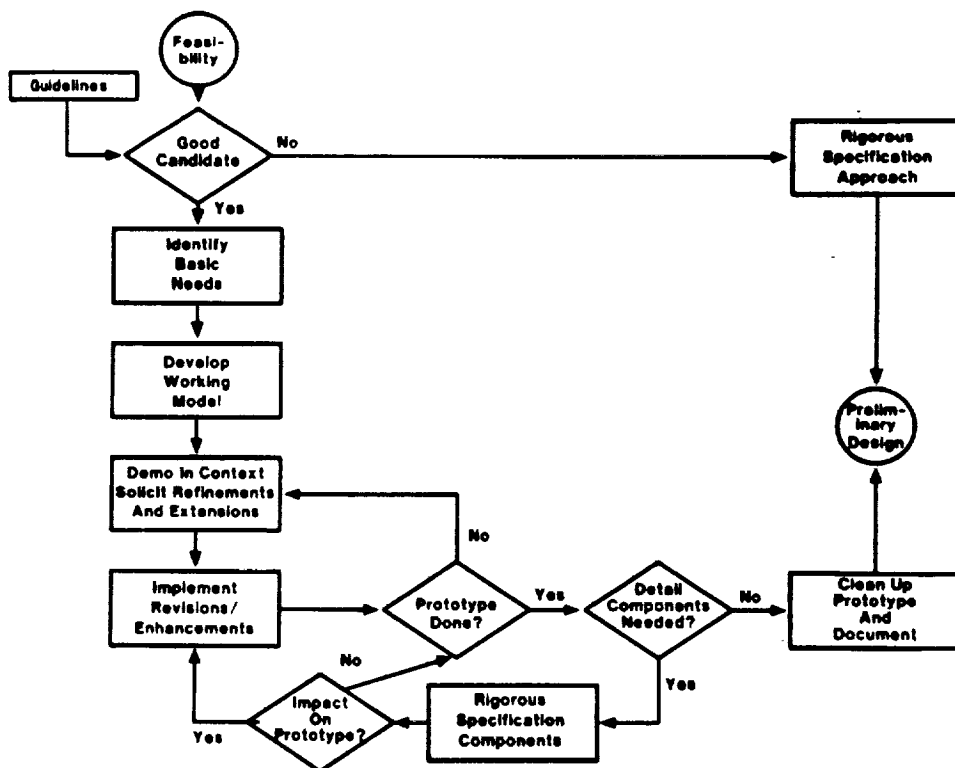
requirements which must be complete, consistent, agreed to, and frozen in place in the later stages of design and development.

An alternative approach to requirements definition is to capture an initial set of needs and to implement, quickly, a prototype system to meet those needs, with the stated intent of iteratively refining the needs and redeveloping the prototype as mutual user/developer understanding of the system needs grows.[2]

Application prototyping refers to a strategy for performing requirements determination wherein user needs are extracted, presented, and developed by building a working model of the ultimate system--quickly and in context. This first cut model (straw-man) serves as a communication anchor between all parties both to enable and enhance a meaningful dialogue. The process is illustrated in Figure 14-1. Prototyping should be considered when:

- Requirements are not specified:
 - Future requirements,
 - Subjective requirements,
 - Fuzzy (probabilistic) requirements;
- Communication gaps exist among the project participants;

FIGURE 14-1: THE PROTOTYPING PROCESS



[Reference 1]

- Extensive iteration is necessary, inevitable, or is being encouraged; and
- An active system model is required.

Prototyping is founded on the belief that people will better understand physical models than logical ones, and will be better able to suggest refinements. It exploits the human fallibilities of the users by placing them in a comfortable and enjoyable role--the wise consumer. Some benefits of prototyping are:

- it accommodates the decision-making and problem-solving styles of the users;
- it requires and increases the amount of active user participation in the development project;
- it provides a vehicle for validating requirements;
- it provides a facility to permit assessment of the impact of the system on the whole user environment;
- it permits early life cycle testing of human/machine interfaces;
- it provides vivid documentation of the Developer;
- it accommodates uncertainty and risk;
- it alleviates project communication problem;
- it permits both forest and tree perspective;
- it provides for high project accountability and visibility;
- it simplifies project management;
- it provides an apprenticeship laboratory; and
- it serves as an interim training vehicle.

14.3 CONSENSUS METHODOLOGIES

Inherent in a consensus methodology are two basic assumptions, namely, that there is the need for consensus and, more importantly, that consensus is a realizable objective.

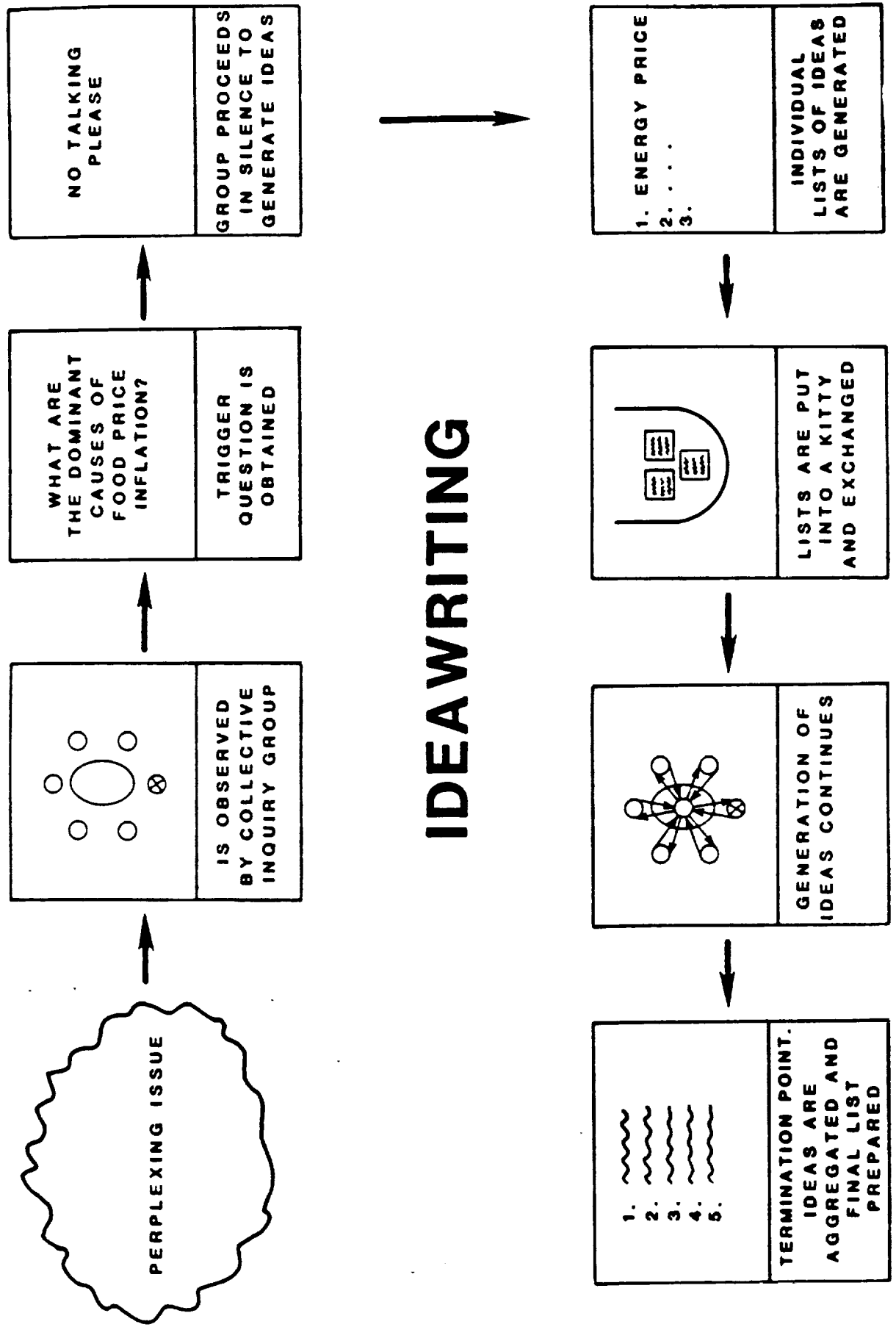
First, inherent in an approach to the resolution of complex problems is the need to apply many minds to it. Second, that the solution, to be enduring, should be understood and supported by implementers.[3]

14.3.1 Ideawriting (Brainwriting)

Ideawriting allows a group to generate ideas quickly and efficiently, while documenting them for further use.[4, PP. 3-4] The process, illustrated in Figure 14-2, has five steps as follows:

- Identify and clarify a "Triggering Question,"
- Silently generate ideas on paper,
- Exchange paper through an "idea pool,"

FIGURE 14-2: IDEAWRITING
[Reference 5]



- Continue generating ideas and exchanging paper, and
- Collect the papers and discuss the ideas generated.

Ideawriting is generally appropriate for all efforts where collective idea generation is of value and especially useful for issue formulation, including problem definition, and identification of objectives.[5, P. 324a]

14.3.2 Nominal Group Technique

Nominal Group Technique (NGT) is an efficient method of generating ideas in groups, for clarifying the generated ideas, for editing the generated ideas, and for developing a preliminary ranking of the set of ideas.[6, Appendix I]. See illustration of the NGT process in Figure 14-3. NGT is generally appropriate for all efforts where collective idea generation is of value and especially useful for issue formulation, including business and government planning and fostering stakeholder participation in planning.[5, P. 325a]. The process includes:

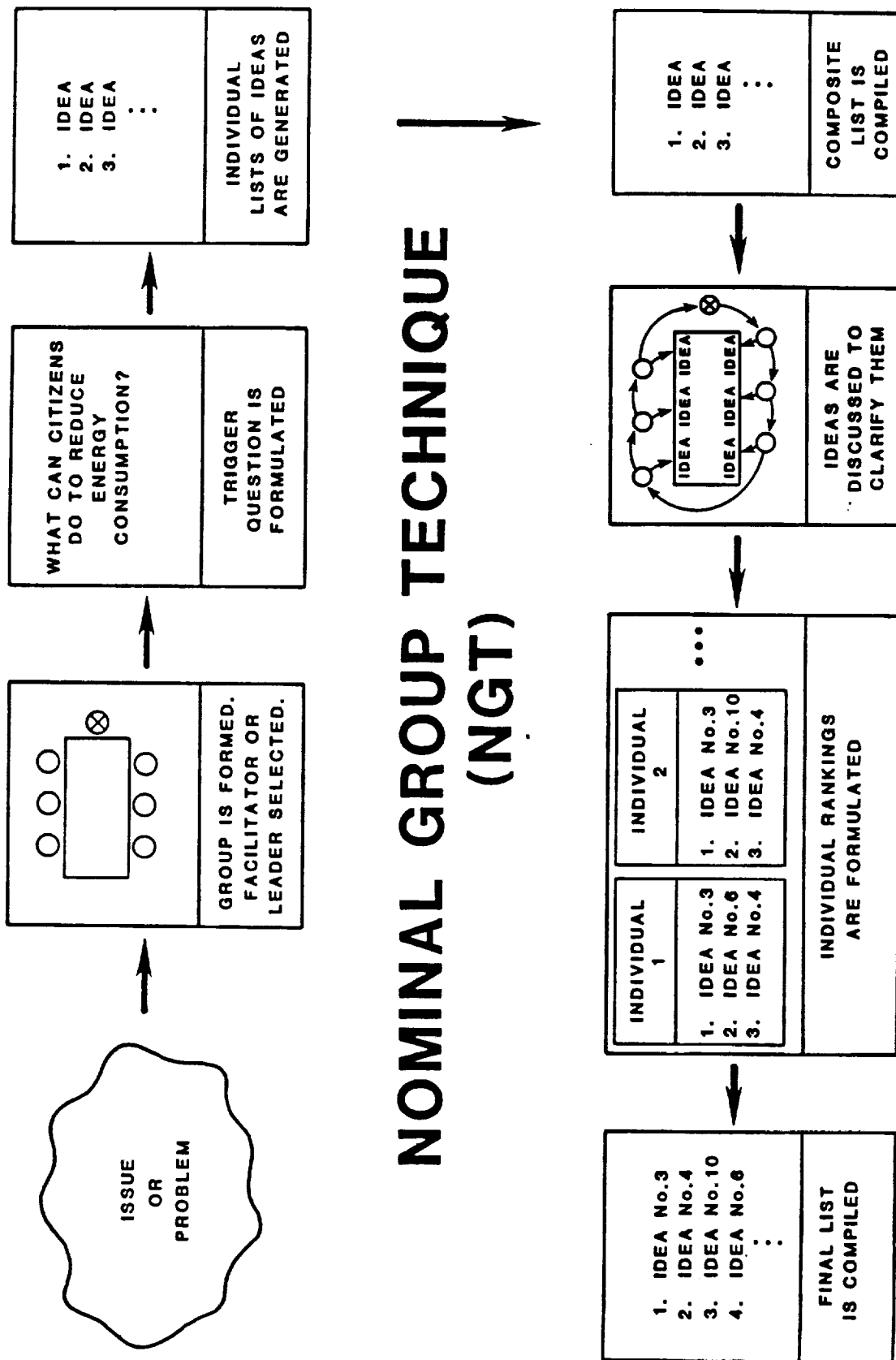
- Organize the group/select the leader,
- Identify and clarify a "Triggering Question,"
- Silently generate ideas on paper,
- Compile ideas,
- Discuss and clarify ideas,
- Rank ideas individually, and
- Compile final list of ideas.

14.3.3 Interpretive Structural Modeling

Interpretive Structural Modeling (ISM) is a computer-assisted learning process that culminates in the development of a structure of an issue, problem, plan, or project. The structure is developed by a group operating with the assistance of a skilled facilitator.[6, Appendix I] ISM is one of the first of the modules in the Interactive Management (IM) chain developed by Warfield. The ISM methodology combines behavioral theory and mathematical theory of relations to organize a set of ideas generated by a group. This is achieved through group participation, with the assistance of a computer.[4, P. 218] The process which starts with a list of elements [from the NGT or Ideawriting techniques] is summarized below and illustrated in Figure 14-4.

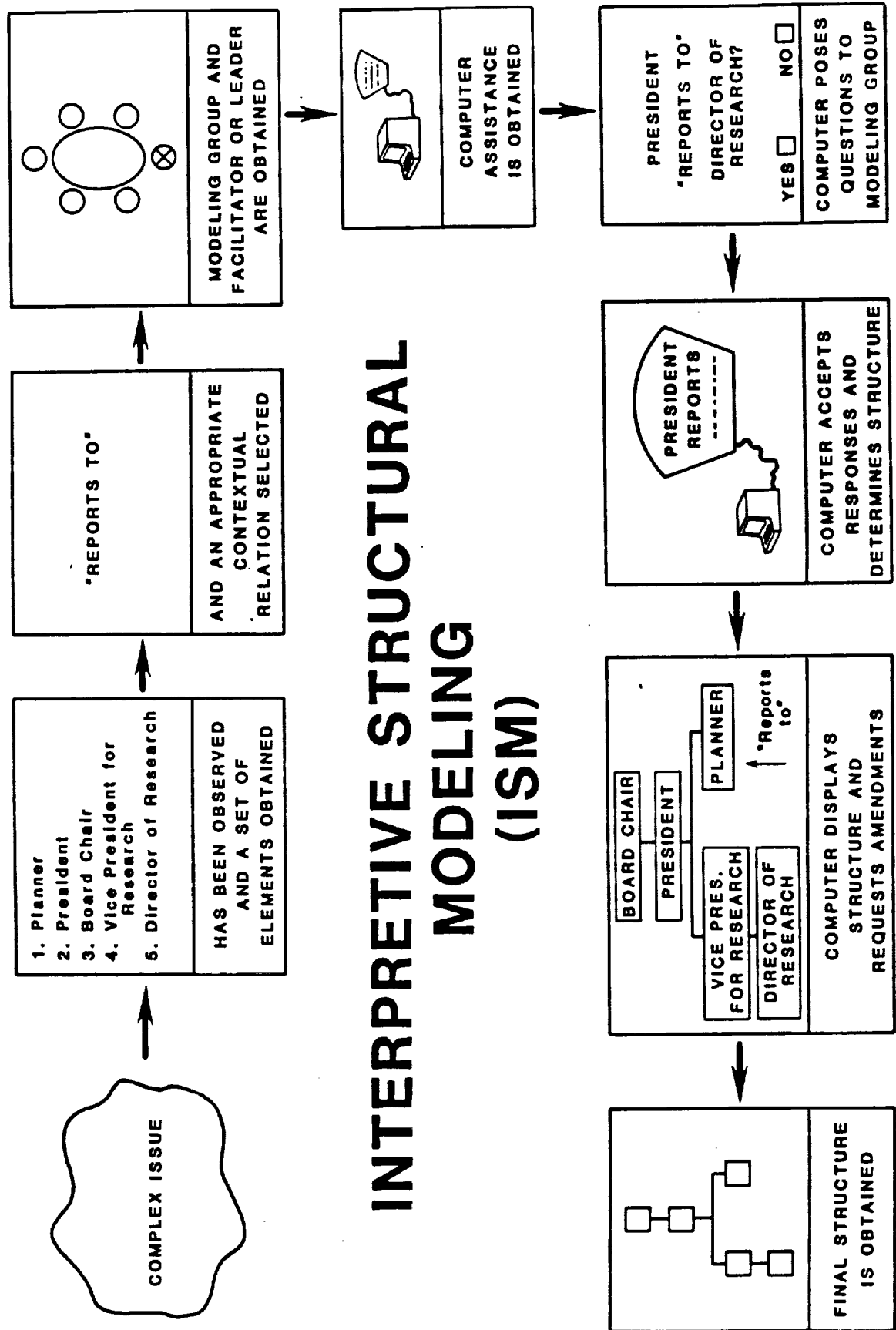
- Start with a list of elements,
- Select an appropriate contextual relationship,
- Organize modeling group and select facilitator/leader,
- Obtain necessary computer systems and support,
- Computer generates and poses questions to modeling group,
- Computer accepts responses and determines structure,

FIGURE 14-3: NOMINAL GROUP TECHNIQUE
[Reference 6]



NOMINAL GROUP TECHNIQUE (NGT)

FIGURE 14-4: INTERPRETIVE STRUCTURAL MODELING



[Reference 6]

- Computer displays structure and requests amendments, and
- Final structure is developed.

14.3.4 Options Field Method

A method for portraying all the conceived dimensions of a prospective design, including the simple options available in each dimension, and showing the clusters of interdependent dimensions. The clusters are arrayed in the sequence with which design choices are to be made, as are the dimensions in each cluster.[6, Appendix I] The process which is summarized below is illustrated in Figure 14-5.

- Identify design group and facilitator/leader,
- Prepare site of design work,
- Generate design options (if necessary, with the aid of Brainwriting or NGT),
- Edit (clarify and simplify) options,
- Sort options into similar set--candidate dimensions, use ISM if necessary,
- Achieve consensus on the dimensions,
- Prepare preliminary options field,
- Identify dimensional clusters,
- Sequence dimensional clusters, and
- Display ordered options field.

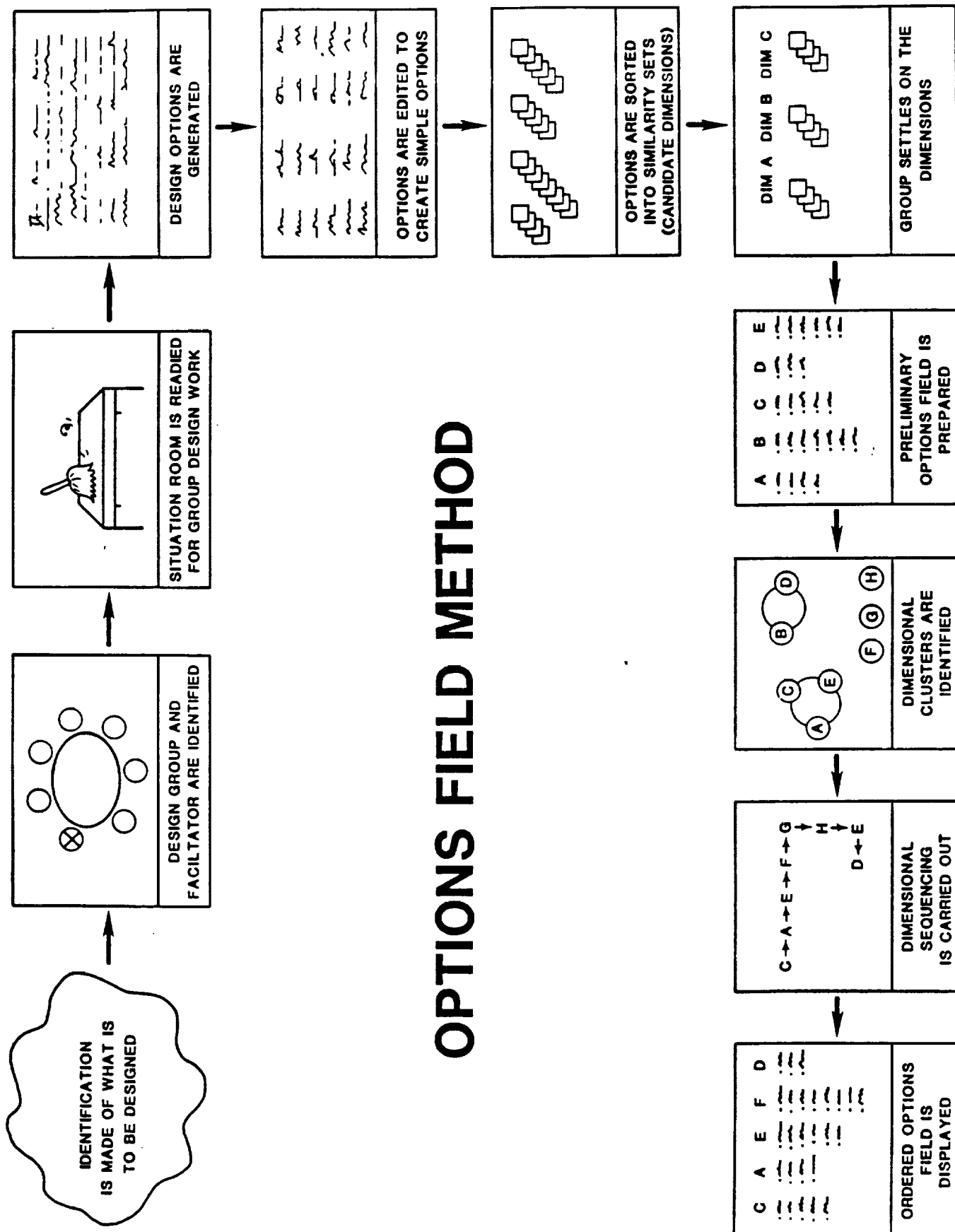
14.4 PROJECT FORCE FIELD ANALYSIS

This method calls for organizing information pertaining to project development into two categories: "forces" that work to restrain change and those that work to facilitate change. See Figure 14-6. In theory, the state of affairs of any situation is allowed to persist because the restraining and facilitation forces are in equilibrium. If the restraining forces should increase, the state of affairs will be worsened; if the facilitating forces are strengthened, the state of affairs will improve.

This scheme of forces is used to determine the best way to introduce change into a situation. A force field analysis begins by identifying all of the restraining and facilitating forces in a situation and their relative strengths. This makes it possible to determine which restraining forces must be weakened or which facilitating forces must be strengthened to move the situation toward the ideal state.

Although the technique was originally proposed as a means for overcoming resistance to change, it can be used by managers in other applications. In project management, the technique can be

FIGURE 14-5: OPTIONS FIELD METHOD

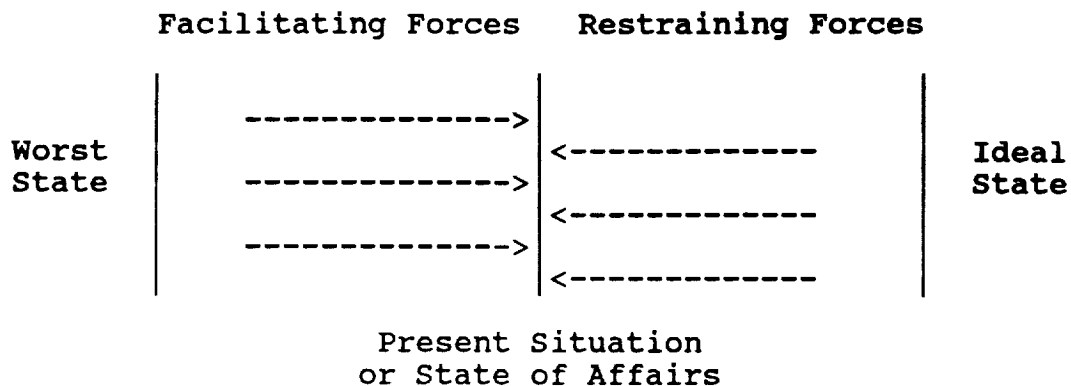


OPTIONS FIELD METHOD

[Reference 6]

used to investigate forces which act on a current project or which might influence an upcoming project, and to determine where emphasis is needed to increase a project's likelihood of success.[7, P. 28]

FIGURE 14-6: FORCE FIELD ANALYSIS STRUCTURE



[Reference 7]

14.5 SUMMARY

The review of methodologies suggests that desirable features from the methodologies being used by other agencies and private enterprises can be incorporated into the ND's methodology to alleviate many of the current deficiencies. The review also identifies the unresolved methodological problems as being primarily in the area of requirements definition, projection/prediction, and communication among the different stakeholders in the problem situation.

The tools and techniques presented in this chapter, in particular, participative design, prototyping and consensus methodologies, have been carefully selected for the potential for ameliorating these outstanding problems.

REFERENCES

2. Boar, Bernard H., Application Prototyping: A Requirements Definition Strategy for the 80s, (New York: John Wiley and Sons), 1984.
3. Ayiku, M.N.B., "Towards Implementing the Menon Report: A Systems Approach Using ISM", Systems Research, Vol. 3, No. 4, 1986, P. 217.
4. "The Future of Volunteers in Human Services: A Commitment to Action", TRW, Inc. and Center for Interactive Management, March 1987.
5. Warfield, John N., A Science of Generic Design, (Textbook in review), 1989.
6. Office of Munitions, Smart Munitions Acquisition Workshops, Final Report, (Washington, D.C.), February 1989.
7. Nicholas, John M., "Successful Project Management: A Force-Field Analysis", Journal of Systems Management, Vol. 40, No. 1, Issue No. 331, January 1989, PP. 24-30.
8. Swift, Michael K., "Prototyping in IS Design and Development", Journal of Systems Management, Vol 40, No. 7, 1989, PP. 14-23.
9. Voltmer, John G., "Selling Management on the Prototyping Approach", Journal of Systems Management, Vol 40, No. 7, 1989, PP. 24-25.
10. Steinblock-Harrison, Teresa, "Techniques and Issues in Rapid Prototyping", Journal of Systems Management, Vol. 36, No. 6, 1985, PP. 8-13.
11. Harrison, Ralph, "Prototyping and the Systems Development Life Cycle", Journal of Systems Management, Vol 36, No. 8, 1985, PP. 22-25.

15. ORGANIZATIONAL CULTURE AND DESIGN PHILOSOPHY

Many writers in the field of systems development stress the need for an organizational culture or a philosophy that is conducive to developing large, complex, systems. Some related values have been extracted from the literature, sorted into general categories, and presented below. In order to avoid the possibility of changing the meaning, through extensive rewriting and interpreting, these value statements have been summarized in the context that they were presented by the original authors.

15.1 WHAT IS MOST IMPORTANT

Johnson Space Center (JSC) nurtures an environment and culture that motivates our people to strive for technical excellence above all else.[1, P. 8]

- Mission success is number one

Space Station Freedom is a multi-year, multi-purpose, international, and evolutionary project. It may be thirty years before the agency can declare it as a total success. Decisions made today will determine tomorrow's successes. Mission success will be measured by a number of parameters, including crew safety, research capability, ease of maintainability, economy of operation, and the ability to evolve to meet future national goals.[2, P. 3]

15.2 THE ROLE OF MANAGEMENT

- Management determines the corporate culture

Corporate culture emanates from the top. It is the top management of the organization that shapes its corporate culture. To change the corporate culture from conservative to progressive, a change is needed in top management.[3, P. 35]

- On-going systems training

The IRS is committed to quality education and training and provides employees with monthly training courses in project management. People cannot be committed to a concept that they do not fully understand. As a result, custom-designed training programs have been developed not only for the employees, but even up through the assistant commissioner levels.[4, P. 11]

- **Effective communication**

The environment and culture at JSC also encourage open, effective communication at all levels on the premise that no surprise is a good surprise when it comes to human-related systems.[1, P. 8]

In many organizations, you get your important points across to top management by telling them rather than writing to them. You learn more about what is really going on by talking to people than by reading reports.[5. P. 23]

- **Managing changing requirements**

If the requirements can be expected to change because the business environment is volatile, the approach used should delay detailed design decisions as long as possible.[6, P. 36]

15.3 ROLE OF THE PROJECT MANAGER

The focus of the project manager and the development team is not to develop a product. The primary responsibility is to satisfy the client.[7]

...., because there is still one person who ultimately must make the decisions and be held responsible for them. Like the controlling stockholder in a large company, the project manager always has 51 percent of the vote.[1, P. 8]

In both NASA and industry, the golden rule applies. The manager with the gold--rules. Make sure you receive and control the money needed to accomplish the mission. If either your boss or your boss's boss controls the money, they in fact control the project. A project manager simply must control all the resources necessary for mission success, or some method of accountability must be devised.[8, P. 19]

15.4 THE ROLE OF THE STAFF

- Every person in the Space Station Freedom organization must think and perform as a systems engineer or manager.

Significant changes in the program can be controlled by the Interface Control Document and Architecture Control Document systems. However, lower level changes are not controlled in this way. These changes require the engineer and manager to think and function as a systems engineer and to question the real effect each minor change has on other elements of the program. This process

is counter to the natural inclination to get the hardware delivered on cost and schedule. The need for this "systems level" consciousness is present in this program more than in any previous NASA program. This management and engineering discipline will be even more necessary as this program continues to develop.[2, P. 5]

15.5 DESIGN PRINCIPLES

- Participative Design

"Participative design means involving users in the design of the system with which they will eventually work. It is involvement of a very positive kind and normally includes an area of joint decision-making, with technical experts acting as advisors.

Participative design can be broad or narrow, tall or flat. That is, it can encompass all decisions from whether to embark on a new strategy or a new technology to the final implementation, or it can be associated with a more limited range of decision making, for example, what needs to be changed and what should go in its place"[9]

- Quality is planned in, designed in, and built in. Quality is not inspected in

Quality starts before designs are drawn and before "metal is bent." The main message here is that each person and organization in the program must understand and believe in the need for quality performance from the onset of the program. Quality encompasses more than just the delivered hardware. It includes management, requirements, design, development, testing, and documentation.[2, P. 3]

- Keep it simple

Engineers have a tendency to make systems more complicated than necessary. Our challenge is to make it simple, there by increasing reliability, minimizing training and crew on-orbit support, and reducing development cost.[2, P. 3]

- Minimize organizational and hardware interfaces, and maximize clear hardware and software accountability

Requirements should be derived, controlled, and accounted for at the appropriate management level. In the space station program, Levels I and II manage the program and develop and manage the requirements. The design and development responsibility, including

requirements verification, are assigned to Level III and the prime contractors.[2, PP. 3-4]

One of the first management decisions the Spacelab Program Office made was to maintain heavy Marshall Space Flight Center (MSFC) engineering involvement from the beginning to the end of the program. This involvement was used to generate and approve all technical requirements in a way that the engineers felt accountable for the technical performance of the Spacelab system even though the overall responsibility resided with the program manager.[10, P. 40]

Contractors are made to share much more responsibility in the design and functioning of "components" and "boxes" that are delivered from one contractor to another. Simply stated, the receiving contractor and the delivery contractor are jointly responsible for the item until the item is fit or functionally demonstrated in the next level of assembly.[2, P. 4]

- Maximize Margins

Add-ons or corrections after the hardware and software are developed are major cost drivers, time wasters, and sources of future problems. Close attention to details in the development phase will save enormous amounts of time and money in the operational phase. The best time to effectively manage resources is early in the program in order to ensure maximum safety, reliability, maintainability, and quality assurance in hardware and software. To over-subscribe such valuable resources as weight, power, volume, and crew time early in the design without the ability for later add-ons will significantly complicate the job.[2, P. 4]

- Maximize redundancy, but also manage it

The space station program has built triple redundancy into critical systems. To extend redundancy further would make the system less manageable. Once backup systems are in place, you have to "manage" them to know you will be able to depend upon second and third levels of redundancy when called upon.[2, P. 4]

- Separation of Requirements from Design:

- System Development Specification (requirements) should describe what is to be done, not how it should be done;
- System Development Specifications (design and plans) specify how requirements should be done;

- D-5000 discourages embedding of design in requirements, when there are requirements which intentionally constrains the design options, they can be listed separately from requirements under a heading "design constraints";[11]
- Meaning of Phase Exit and of Baselines:
 - Phase exit corresponds to a review board decision that (1) the information in the SDS has reached the Baseline specified for that phase, and that (2) the developing organization is ready to begin executing the actions defined for the next phase;
 - Phases are named for the dominant activity that occurs in each phase. This naming strategy does not imply that activities such as planning, requirements, and design are constrained to occur in only one phase. In fact, D-5000 assumes the opposite is true: that planning, requirements, and design milestones and activities occur in every phase.
 - The concept of phases in the D-5000 model refers to the maturity and degree of detail of the data items--an iterative process. This is different from the waterfall model which generates new sets of data items in phase--a sequential process;[11]
- Early Involvement of Test and Operations:
 - D-5000 specifies full Test and Operations Engineering participation during the early development phases;
 - Test and operations resource requirements (i.e., budgets and schedules) should be understood before approving plans;
 - Requirements addressing system testability and operability should be defined before approving requirements;
 - Cost and schedule impacts of testing system requirements should be evaluated at the same time as implementation costs during the requirements scrubbing process;
 - System design trade studies should be evaluated in terms of life cycle costs (e.g., including operations costs), not just development costs;[11]

- **Separation between Architectural and Implementation Design:**
 - D-5000 advocates that architectural design should be functional, that is, "designed or developed chiefly from the point of view of use;"[Webster]
 - D-5000 advocates that implementation design should be physical, that is; "of, relating to, or according with material things;"[Webster]
 - Architectural designs that are implementation dependent should be avoided. Rather, architectural design should permit analysis and selection of an implementation design from among many viable alternatives; and
 - This principle of separating architectural design from implementation design is particularly important for systems that are expected to change with evolving technology.[11]
- **Functional Partitioning:**
 - D-5000 advocates that higher level systems be partitioned into subsystems by function, not by software, hardware, and/or operations;
 - D-5000 discourages premature assignment of functions to hardware, software, people, and procedures; and
 - D-5000 advocates performing implementation media trade studies, analysis, and decisions as part of lower-level design, where the appropriate expertise resides.[11]
- **Alternative Designs/Trade Studies:**
 - The D-5000 model specifies that the design process should generate more than one response to requirements. The design process should generate alternative designs, develop and apply selection criteria, and perform trade studies; and
 - The D-5000 model treats plans similarly.[11]

15.6 SUMMARY

While some of the values are conflicting, establishing a set of values for the ND will provide the project manager and systems engineer with a very basic, unchanging, frame of reference which (s)he could rely on should the project management/systems

engineering methodology fail. Much like navigators relying on the sun or the magnetic north pole when local navigational systems fail.

REFERENCES

1. Cohen, Aaron, "Project Management: JSC's Heritage and Challenge", Issues in NASA Program and Project Management, (Washington, D.C.: Office of Management, NASA), 1989, PP. 7-16.
2. Odom, James B., "Guiding Principles for the Space Station Program", Issues in NASA Program and Project Management, (Washington, D.C.: Office of Management, NASA), 1989, PP. 3-7.
3. Khan, M. B. and Martin, Merle P., "Managing the Systems Project", Journal of Systems Management, (Association for Systems Management), Volume 40, No. 1, Issue No. 331, January 1989, PP. 31-37.
4. Kerzner, Harold, "Systems Project Management: A Case Study at the IRS", Journal of Systems Management, (Association for Systems Management), Volume 40, No. 1, Issue No. 331, January 1989, PP. 7-9.
5. Newman, Thomas C., "Controlling Resources in the Apollo Program", Issues in NASA Program and Project Management, (Washington, D.C.: Office of Management, NASA), 1989, PP. 23-26.
6. Dos Santos, Brian L., "A Management Approach to Systems Development Projects", Journal of Systems Management, (Association for Systems Management), March 1986, PP. 35-41.
7. Lackman, Michael, "Controlling the Project Development Cycle", Journal of Systems Management, (Association for Systems Management), February 1987, PP. 7-29.
8. Guastaferro, A., "Shared Experiences in NASA Projects", Issues in NASA Program and Project Management, (Washington, D.C.: Office of Management, NASA), 1989, PP. 17-22.
9. Mumford, E. and Henshall, D., "A Participative Approach to Computer Systems Design", Associated Business Press, 1978.
10. Lee, Thomas J., "Evolution of a Technical Management Organization", Issues in NASA Program and Project Management, (Washington, D.C.: Office of Management, NASA), 1989, PP. 37-42.

11. SSORCE, The JPL System Development Management Guide: Working Draft, (Pasadena, California: Jet Propulsion Laboratory, California Institute of Technology), February 1989.

16. CONCLUSIONS AND RECOMMENDATIONS

The study conducted in Phase One of the project identified, by surveying users, several problems with the Systems Management Policy--the document which describes the systems engineering methodology being used by the ND.

These problems were of two general types--deficiencies in the methodology and problems with the application and/or management of the systems engineering process. These problems are summarized as follows:

(1) Deficiencies in the Systems Engineering Methodology being used by the ND

- While the SMP suggests that it can be tailored to meet the needs of different projects (types and sizes), in practice, considerable effort is required to streamline it for the very small projects, and the feasibility of such streamlining has been questioned by some project managers.
- The details (steps, tasks, activities) of the methodology are not clearly defined.
- While the methodology specifies the type of documents that should be produced at different points in the system development cycle, it does not provide sufficient details about the content and structure of those documents.
- The methodology provides no assistance with projecting or predicting future requirements.
- The methodology does not respond adequately to changing and emerging requirements. Thus, at the time the system is implemented it is usually responding to requirements of several years earlier, not the current requirements.
- The methodology does not support the development of systems in situations where it is not possible to define the requirements [institutional systems or systems on the cutting edge of technology].
- The methodology provides minimal support with tools and techniques for communicating among participants on a major project, e.g., graphics and prototyping.
- The methodology is not sufficiently flexible with regard to scheduling of tasks to accommodate changing priorities and funding levels and to maximize the effective utilization of resources.

- The methodology does not address adequately the possibility of extensively redesigning or modifying an existing system to incorporate new requirements and capabilities and extend its useful life, as opposed to retiring that system and developing a completely new system to replace it. Modifying an existing system tends to shorten the time to have a capability in place.

(2) Problems with the Application and Management of Systems Engineering within the ND

- While the ND and the MO&DSD frequently work with other directorates and major organizational units within NASA on large agency-wide projects, each organizational unit uses its own methodology, with the project manager having responsibility for coordinating and negotiating approaches.
- While proposals are reviewed extensively for adherence to the requirements of the RFP, some staff members are concerned that methodology is not given adequate importance among the evaluation criteria and in the review and evaluation process.
- Systems engineering support contractors are generally involved in routine systems analysis work, and they are usually not used effectively for systems engineering management or in supporting the application of systems engineering methodologies to major ND projects.
- While project management plans are reviewed administratively, a concern of some staff members is that they are not reviewed rigorously from a systems engineering perspective.
- Some of the Division's smaller projects, the sustaining engineering projects, are not developed with a formal systems engineering methodology.

16.1 CONCLUSIONS

A review of the literature, conducted in this phase of the project, revealed that other organizations which developed large systems had similar concerns and were encountering similar problems.

The focus of this phase of the project has been on identifying and evaluating methodologies with the potential for resolving the methodological deficiencies identified in the SMP. Six methodologies were preselected, because they were being used by private enterprises and government agencies which had systems development problem similar to the ND.

The methodologies reviewed showed significant differences in structure. At one extreme, the SMP has eleven phases and a WBS of eight pre-defined items. At the other extreme, several methodologies had only five phases and no pre-defined work breakdown structure. Some methodologies recommend a hierarchical decomposition of the work into functions and or subsystems and components.

Having provided these basic structures, some methodologies describe the meaning of the phases and the other structural divisions while other methodologies provided far greater details on: inputs to the phases, tasks to be performed in the phases, documents to be developed, criteria for exiting, and outputs of the phases.

All methodologies start generally with some definition of requirements. It seems clear that some take more responsibility for that activity than others. For example, some methodologies treat requirements as inputs to the process which should be verified as opposed to having responsibility for developing and defining such requirements.

There is very little consistency from one methodology to another in terms of the partitioning of the system development cycle into phases or the activities to be performed within these phases.

The methodology being used by DSMC provide an extensive discussion of modification of systems that have been implemented. This is different from the others which plan for the operation, but terminate the systems engineering activities upon delivery or installation and testing of the system.

Systems engineering is presented within the more general context of project management or systems development, where a program or project manager, with ultimate responsibility for the project or system oversees the systems engineering activities.

The methodologies reviewed took a variety of approaches in specifying the tasks of the systems engineer. At one extreme there is an overall description of systems engineering as a work breakdown item. At the other extreme, the role of the systems engineer was described for each phase of the cycle. Somewhere between both extremes is a description of the systems engineering as an iterative set of activities to be conducted at different points in the systems development cycle.

Some methodologies describe systems engineering as comprising a management function, which supervises all technical inputs to the project and a technical function which conducts the systems engineering process or technical activities. Other methodologies

seem not to recognize the management function of systems engineering.

All of the methodologies advocate incremental development and testing as opposed to the "big bang", where all functions and capabilities of a complex system are developed concurrently and incorporated simultaneously. This generally involves baselining--establish a minimum capability (functionality) of the system for initial installation.

All of the methodologies recognize the potential for improving the system by iterating, either within phases of the development cycle or through sequential phases of the cycle.

While each methodology has some unique strengths, no individual methodology will satisfy all of the needs of the ND. Thus, the process of developing methodologies that are more suitable for the ND should involve extracting desired features from a variety of sources, integrating, and testing to verify that they will satisfy the requirements. The methodologies reviewed can improve on the methodology in use by the ND in the following areas:

- partitioning the project into phases, work breakdown items, tasks, functions, subsystems, and work packages which can be used in planning the project;
- handling new information, feedback, or unforeseen circumstances;
- acquisition by procurement;
- scheduling of project resources;
- identifying and selecting human and material resources;
- specifying the documentation needed at different points in the development process;
- management procedures to ensure that the methodology is being applied as intended;
- ways of addressing critical considerations such as national security, and risk to humans and the environment;
- clearer, more precise, and more complete documentation;
- techniques for retaining key personnel on the project; and
- tools and techniques for redesigning and modifying to add capabilities and extend the useful life of the system.

Needs of the ND that cannot be satisfied by the methodologies reviewed include:

- improving the process of scoping the basic methodology to projects of different types and sizes;
- tools and techniques for predicting or projecting future requirements;
- strategies and techniques for designing and developing systems in the absence of specific requirements; and
- tools and techniques, including graphics and prototyping, for communicating among individuals and organizations working on major systems projects.

16.2 RECOMMENDATIONS

This study recommends the following:

- (1) That the ND develop a statement of policy and related design principles to guide the project manager, systems engineering manager, and staff in areas where the methodology fails to provide adequate guidance.
- (2) That the ND consider training in project management and systems engineering to be an on-going requirement for successful management of large systems development projects.
- (3) That the ND resolve the weaknesses in the SMP by incorporating desirable features from the other methodologies reviewed in this effort and from other sources. Table 13-1 shows the methodologies that are judged to be superior to the SMP in the different problem areas.
- (4) That the ND incorporate more participative design, prototyping, and consensus building techniques in its systems development process to provide some relief in the difficult and yet unresolved problem areas of communication among stakeholders (design team: contractors and staff, users, managers, etc.), and definition, projection/prediction of requirements.

APPENDIX A: LIST OF SPEAKERS
SPRING SEMESTER, 1989

<u>Date</u>	<u>Speaker/Topic</u>
January 24	Peter Freeman, National Science Foundation " Systems Engineering: Approaches to Increase productivity"
February 2	Warren Falconer, AT&T Bell Laboratory "Future Trends in Systems Engineering"
February 6	Andrew Sage, George Mason University "Systems Engineering Methodology"
February 7	Izeller Cureton-Snead, Jet Propulsion Laboratory "Systems Engineering in a Dynamic Environment"
February 16	John Carraway, Jet Propulsion Laboratory "Systems Engineering Methodology at Jet Propulsion Laboratory"
February 28	John Warfield, George Mason University "Systems Engineering: Large-Scale Systems Design"
March 6	Wolter Fabrycky, Virginia Polytechnic Institute "Systems Engineering Methodology-Life Cycle Cost & Planning"
March 16	King Eng, Howard University "Systems Engineering Practice"
March 20	Bernard Rudwick, Defense Systems Management College, "The Role of Systems Analysis in Controlling Development Programs"
April 14	Thomas Allen, Massachusetts Institute of Technology "Complex Matrix-Structured Management"
April 17	John Derbinger, HUGHES Research Laboratory " Systems Engineering"
May 2	Arnold Ruskin, JPL Consultant " Systems Engineering Methodology"